AD-A039 841 NAVAL RESEARCH LAB WASHINGTON D C F/G 17/9

SURVEILLANCE RADAR SYSTEMS EVALUATION MODEL (SURSEM) HANDBOOK.(U)

JAN 77 D J KAPLAN, A GRINDLAY, L DAVIS

UNCLASSIFIED NRL-8037 NL

1 OF 2

ADA039 841

NRL Report 8037

Surveillance Radar Systems Evaluation Model (SURSEM) Handbook

DAVID J. KAPLAN, ALEX GRINDLAY, AND LAURA DAVIS

Radar Analysis Staff
Radar Division

January 14, 1977

NAVAL RESEARCH LABORATORY
Washington, D.C.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: NRL ~~Report~~ 8037

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): SURVEILLANCE RADAR SYSTEMS EVALUATION MODEL (SURSEM) HANDBOOK.

5. TYPE OF REPORT & PERIOD COVERED: Final report.

6. PERFORMING ORG. REPORT NUMBER:

7. AUTHOR(s): David J. Kaplan, Alex Grindlay, ~~and~~ Laura Davis

8. CONTRACT OR GRANT NUMBER(s):

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
Naval Research Laboratory
Washington, D.C. 20375

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
62712N SF12-141-401 NISC
PO18083 NRL Nos. R12-06.801
and R12-38.401

11. CONTROLLING OFFICE NAME AND ADDRESS:
NAVSEASYSCOM
NISC

12. REPORT DATE: January 14, 1977

13. NUMBER OF PAGES: 167

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office):

168p.

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE:

16. DISTRIBUTION STATEMENT (of this Report):

F12141

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report):

18. SUPPLEMENTARY NOTES:

19. KEY WORDS (Continue on reverse side if necessary and identify by block number):

Surveillance	Detection
Radar	Jamming
Computer model	Propagation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number):

The Surveillance Radar Systems Evaluation Model (SURSEM) is a user-oriented computer model designed to generate radar single-scan and cumulative probability of detection values as a function of target range and orientation within a specified environment. A model overview, instructions for the potential SURSEM user, and a detailed description of each SURSEM subroutine have been developed as well as a FORTRAN program listing and a description of MTIMOD, a moving target indicator signal-processing subprogram to be integrated into SURSEM.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

251 950

 SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

SURVEILLANCE RADAR SYSTEMS EVALUATION MODEL (SURSEM) HANDBOOK

1 INTRODUCTION

1.1 Background

This handbook constitutes the documentation for the Surveillance Radar Systems Evaluation Model (SURSEM), developed by the Radar Analysis Staff (Code 5308) of the Naval Research Laboratory. This documentation has been sponsored by NAVSEA-SYSCOM and NISC, with the objective of providing the Navy with a fully described, user-oriented computer model of a surveillance radar.

1.2 Organization

This handbook serves two basic purposes: it reflects the current capability of the computer model, and it provides the potential user with instructions for its independent use. The handbook is divided into four sections. The first section, the introduction, includes a general description of the model, followed by comments concerning its growth potential. Instructions for the user make up the second section, which gives a complete description of the input parameters and printed output. This section also includes a test case that is explained in detail. The third section contains the documentation of the individual computer subroutines. Such documentation consists of an analytical description, macro and micro flowcharts, a table of FORTRAN variables together with a description and units used, and a FORTRAN listing for each program subroutine. The fourth and last section consists of two appendixes. Appendix A is a FORTRAN listing of SURSEM as used on a PDP DEC-10 time-sharing system. Appendix B is a description of MTIMOD (Moving Target Indicator Modification), a subprogram designed to use MTI signal-processing techniques to calculate the signal-to-noise ratio in a rain and chaff environment. MTIMOD has been programmed but has not yet been fully tested or incorporated into SURSEM.

1.3 Model Overview

SURSEM produces radar single-scan and cumulative probability-of-detection values as a function of target range and orientation. The radar operates within a specified scenario, defined by the user to include the target to be detected, up to nine additional sources of jamming radiation, and an optional environment of wind, rain, and multipath. An engagement (in the limited sense of a target flying into the detection range of the stationary surveillance radar along a predetermined course and the radar attempting to detect the target during its flight) is initiated at the time the target leaves its initial position and is terminated when the target reaches its final position.

Manuscript submitted July 14, 1976.

SURSEM has been constructed as a modified time-step model. The time steps involved are determined by the elapsed time between radar scans illuminating the target. The surveillance radar under examination is characterized by its radar scan modes. A radar scan mode is a device defining radar operating characteristics for the illumination of a specific geometrical region. Typical radar scan modes include long-range search; high-angle, low-energy search; burnthrough; and horizon scan. At the onset of the engagement (i.e., when the target leaves its initial position), the time at which each operational radar scan mode will first illuminate the target is determined. The mode corresponding to the smallest time is selected as the initial scan mode, and its time of occurrence becomes the current engagement time.

The position and orientation of the target at the current time are calculated under the assumption that the target follows a linear flightpath at constant speed. The positions and orientations of any other defined targets, which are treated as additional sources of jamming radiation, are determined in the same manner.

The probability that the radar will detect the target during the selected radar scan is calculated by means of the user-designated Marcum-Swerling cross-section model [1]. SURSEM is concerned with the evaluation of *surveillance* radars and has no provision for target tracking. Scan-to-scan independence is assumed in the calculation of the cumulative probability of detection.

1.4 Future Growth

SURSEM will change and expand as the result of application to an increasing variety of problems. Some areas of interest for the future modification of SURSEM have already been identified and include the following:

1. Signal processing—integration of MTIMOD into SURSEM as mentioned above, as well as the development and incorporation of other signal-processing options, such as pulse-compression and pulse-doppler techniques,
2. Detection confirmation—greater detail in the modeling of the process that confirms the detection of a potentially detectable target,
3. Area clutter—more faithful rendition of the effects of sea state, shadowing, and polarization on the area-clutter return over a wide range of pulse widths,
4. Refractive-index models—inclusion of an exponential refractive-index model,
5. Point signals—development of an empirical point-scatterer model and modeling of polarization effects,
6. Volume clutter—inclusion of a chaff model,
7. Antenna pattern—improved sidelobe modeling and computation of the horizontal gain pattern factor when considering target detection with discrete-beam-forming radar.

Modifications of the model will be reflected in revisions to this handbook. Revised pages and new sections will be sent to any user who *requests* this information.

The authors of this handbook are available on a limited basis for consultation on problems related to the compilation and execution of the model. We are also interested in negotiating with potential sponsors for the development of the model's growth potential and in some aspects of performing analyses of radar systems through the application of the model.

1.5 Acknowledgments

Large portions of the sea-clutter and multipath routines in SURSEM were programmed by J. D. Wilson at NRL. O. F. Forsyth of the Stanford Research Institute contributed to the development of early versions of the model.

2. INSTRUCTIONS FOR THE USER

2.1 Input

SURSEM is designed to generate radar single-scan and cumulative probability-of-detection values as a function of target range and orientation in a specified environment. An engagement scenario consists of a radar, one target to be detected, and up to nine additional sources of jamming radiation set in a specified environment. The required input information is divided into the definitions of radar, targets, and an environment, together with several output control parameters. Each of these is considered in detail.

2.1.1 *Radar*

A radar is described by specifying 11 basic parameters followed by 16 descriptors for each of up to 15 operational radar scan modes. Typical radar scan modes include long-range search; high-angle, low-energy search; burnthrough; and horizon scan. The 11 basic radar input parameters are—

1. Radar frequency in megahertz
2. Antenna pattern indicator (0 or 1, where 0 = pencil beam and 1 = cosecant-squared beam)
3. Receiver noise in dB
4. Horizontal 3-dB beamwidth in degrees
5. Vertical 3-dB beamwidth in degrees
6. One-way antenna gain in dB
7. One-way sidelobe level in dB down from peak
8. Receiver line loss in dB
9. Transmitter line loss in dB
10. Number of scan modes (to be) defined, not to exceed 15
11. Linear polarization in degrees, from 0 to 90, where 0° = horizontal and 90° = vertical

Each radar scan mode (as specified in item 10 above) is described by the following 16 input parameters:

1. Lower limit of elevation-angle coverage in degrees
2. Upper limit of elevation-angle coverage in degrees
3. Peak power in megawatts
4. Pulse length in microseconds
5. Interlook period (time between scans) in seconds
6. Scan offset (relative to radar initialization) in seconds

7. Instrumented range in nautical miles
8. Mode-dependent loss in dB
9. Number of pulses integrated
10. Minus $\log_{10}$ (false-alarm probability); for example, if the false-alarm probability is 10^{-6}, it would be entered here as 6
11. Compressed-pulse length in microseconds
12. Sea clutter improvement factor in dB
13. Intermediate-frequency bandwidth in megahertz; if 0 is entered, bandwidth is set at 1.0/(compressed-pulse length)
14. Mode-dependent frequency increment in megahertz; if nonzero, the effective horizontal and vertical beamwidths and antenna gain for this scan mode are also affected (see description of the SURSEM Executive Routine)
15. Blanking time in microseconds; if 0 is entered, blanking time is set at pulse length
16. Rain-clutter improvement factor in dB

Radar scan modes are numbered in ascending order as they are defined, beginning with number 1.

The location of the radar platform, which is assumed to remain stationary throughout an engagement, and the antenna height above sea level are specified by three radar position input parameters (in thousands of feet):

1. x position coordinate
2. y position coordinate
3. z position coordinate (antenna height above sea level)

It is often convenient to let the radar platform be located at the origin (0,0) of the scenario coordinate system.

2.1.2 *Targets*

The model can accommodate from one to nine moving objects called targets; however, only the primary target (the first target defined) is used in the calculation of the radar detection-probability values. Additional targets can be defined as jammers. Each target is described by the following 13 input parameters:

1. x coordinate
2. y coordinate
3. z coordinate

(items 1–3: initial target position in thousands of feet)

4. Time in seconds at which target leaves its initial position

5. x coordinate
6. y coordinate } terminal target position in thousands of feet
7. z coordinate
8. Time in seconds at which target reaches its terminal position
9. Head-on radar cross section in square meters
10. Broadside radar cross section in square meters
11. Minimum radar cross section in square meters
12. Jamming power density in watts per megahertz
13. Marcum-Swerling cross section model number (0, 1, 2, 3, or 4)—see discussion of Subroutine MARSWR for a description of cross-section models

All targets are assumed to follow constant-speed, linear flight paths calculated from the initial and final positions and times specified above.

2.1.3 *Environment*

The effects of wind, rain, and multipath on the detection capability of the radar are included in the model through the specification of the following environmental input parameters:

1. Wind speed in knots
2. Height of wind-speed measurement in thousands of feet
3. Multipath indicator (0 or 1, where 0 = no multipath and 1 = multipath)
4. Rainfall rate in millimeters per hour

The wind speed and the height at which it is measured are used to determine the sea state in the calculation of the sea-clutter return. The rainfall rate permits the calculation of the rain-attenuated signal energy and the contribution of rain backscatter to the total noise energy.

2.1.4 *Output Control*

The output control parameters determine the amount of information to be printed as the result of each engagement and select the radar scan modes (from those defined as radar inputs) to be considered for use during the engagement. A line of output data (discussed in the following section) can be printed for each radar scan performed during which the target was within the instrumented range of the radar.

The six output control input parameters are as follows:

1. Lower bound for which single-scan probability of detection will be printed.
2. Upper bound for which single-scan probability of detection will be printed.

3. Largest cumulative probability of detection to be printed. When the cumulative probability exceeds this value, no more output will be printed for the engagement.

4. Line print indicator. For example, if this value were 5, then every fifth line of generated output would be printed. The option to suppress *all* printed output is available prior to scenario input, as noted in Example 1.

5. Total number of radar scan modes (from those defined earlier during radar input) to be used during the engagement. If all scan modes defined are to be used, a zero may be entered here and a single number (dummy) for the next parameter.

6. Actual radar scan mode numbers to be used during the engagement, in ascending order. The first scan mode defined is mode 1, the second mode defined is mode 2, etc. This option allows a series of engagements to be performed using a number of scan mode combinations without redefining radar inputs for each run. See Example 1 for details.

2.2 Output

An engagement begins at the time the primary target leaves its initial position. From this time until the time the target reaches its terminal position, the model considers each radar scan mode and corresponding scan time in turn and selects for operation the mode that will next illuminate the target. For each scan mode selected, the following 15 output values, constituting a line of output data, are generated:

Output	*Description*
INDEX	Scan counter
MODE	Scan mode number
TIME	Time of scan occurrence in seconds (relative to engagement initialization at time 0)
RANGE	Slant range of target from radar in thousands of feet
BV	Elevation angle of target in degrees
BH	Azimuth angle of target in degrees
SIGMA	Target cross section in square meters
FCTR	Multipath pattern-propagation factor in dB (calculated by subroutine MULPTH)
ESIG	Signal energy in dB (with respect to a joule)
NAMB	Ambient noise in dB (with respect to a joule)
NCLT	Clutter energy in dB (with respect to a joule)
NJAM	Jamming energy in dB (with respect to a joule)
E/N	Signal energy-to-noise energy ratio in dB
PDSS	Single-scan probability of detection
PDCUM	Cumulative probability of detection

The sequential generation of output data for each occurrence of selected scan modes terminates when the target reaches its final position, thereby ending the engagement.

2.3 Example

SURSEM has been used at the Naval Research Laboratory both in batch processing on a CDC-3800 computer and on several time-sharing systems, including the CDC KRONOS and an in-house PDP DEC-10 system. This example is directed toward the time-sharing version of SURSEM. The difference between the batch and time-sharing versions is in the input-output format structures. Modifications applicable to the operation of the batch version are noted in the next section, "Batch Processing." Appendix A contains a listing of the time-sharing version of SURSEM.

Example 1

```
1. THIS IS SURSEM, THE NRL SURVEILLANCE RADAR SYSTEM EVALUATION MODEL.
2. DO YOU NEED A DESCRIPTION OF THE INPUT VARIABLES?  NO
3. DO YOU WANT DETAILED OUTPUT?  YES
4. TYPE 11 BASIC RADAR INPUTS:  3500 1 5 1.7 1.3 39.7 25 1.5 2 1 0
5. TYPE 16 INPUTS FOR MODE  1:  -.45 .85 2.5 12 6 0 240 3 1 6 .05 0 20 0 0
   0
6. TYPE SHIP/RADAR POSITION COORDINATES (X,Y,Z) IN KFT:  0 0 .1
7. TYPE NUMBER OF TARGETS:  1
8. TYPE 13 PARAMETERS FOR TARGET  1:  240 0 .1 0 30 0 .1 720 1 1 1 0 0
9. TYPE 4 ENVIRONMENTAL PARAMETERS:  10    1 1 1
10. TYPE OUTPUT CONTROL PARAMETERS:  0 1 1 4 1 1

11. INDEX        TIME          BV        SIGMA      ESIG      NCLT      E/N      PDCUM
12.        MODE        RANGE        BH         FCTR      NAMB      NJAM      PDSS

13.    1         336.0         0.01         1.0      -187.     -350.    -20.50     0.000
14.          1          142.0        0.0        -34.     -199.     -350.      0.000

15.    1         360.0         0.01         1.0      -180.     -350.    -14.47     0.000
16.          1          135.0        0.0        -28.     -199.     -350.      0.000

17.    1         384.0         0.02         1.0      -174.     -350.     -8.42     0.000
18.          1          128.0        0.0        -23.     -199.     -350.      0.000

19.    1         408.0         0.03         1.0      -167.     -350.     -2.34     0.002
20.          1          121.0        0.0        -17.     -199.     -350.      0.000
```

Example 1 Continued

```
21.  2        432.Ø     Ø.Ø4      1.Ø    -16Ø.    -35Ø.     3.77    Ø.ØØ3
22.       1        114.Ø     Ø.Ø       -11.    -199.    -35Ø.    Ø.ØØ2

23.  6        456.Ø     Ø.Ø4      1.Ø    -154.    -35Ø.     9.93    Ø.312
24.       1        1Ø7.Ø     Ø.Ø        -6.    -199.    -35Ø.    Ø.233

25.  1Ø       48Ø.Ø     Ø.Ø5      1.Ø    -147.    -35Ø.    16.Ø8    1.ØØØ
26.       1        1ØØ.Ø     Ø.Ø        -Ø.    -199.    -35Ø.    1.ØØØ

27.  14       5Ø4.Ø     Ø.Ø6      1.Ø    -141.    -35Ø.    21.46    1.ØØØ
28.       1         93.Ø     Ø.Ø         4.    -199.    -35Ø.    1.ØØØ

29.  18       528.Ø     Ø.Ø6      1.Ø    -136.    -35Ø.    25.18    1.ØØØ
30.       1         86.Ø     Ø.Ø         8.    -199.    -35Ø.    1.ØØØ

31.  22       552.Ø     Ø.Ø7      1.Ø    -133.    -35Ø.    27.74    1.ØØØ
32.       1         79.Ø     Ø.Ø         9.    -199.    -35Ø.    1.ØØØ

33.  26       576.Ø     Ø.Ø8      1.Ø    -131.    -184.    28.47    1.ØØØ
34.       1         72.Ø     Ø.Ø         9.    -199.    -35Ø.    1.ØØØ

35.  3Ø       6ØØ.Ø     Ø.Ø9      1.Ø    -133.    -182.    25.95    1.ØØØ
36.       1         65.Ø     Ø.Ø         6.    -199.    -35Ø.    1.ØØØ

37.  34       624.Ø     Ø.Ø9      1.Ø    -143.    -181.    14.56    1.ØØØ
38.       1         58.Ø     Ø.Ø        -6.    -199.    -35Ø.    Ø.992

39.  36       648.Ø     Ø.1Ø      1.Ø    -146.    -179.    1Ø.34    1.ØØØ
40.       1         51.Ø     Ø.Ø       -12.    -199.    -35Ø.    Ø.327

41.  4Ø       672.Ø     Ø.11      1.Ø    -123.    -177.    31.93    1.ØØØ
42.       1         44.Ø     Ø.Ø         8.    -199.    -35Ø.    1.ØØØ

43.  44       696.Ø     Ø.12      1.Ø    -122.    -174.    32.28    1.ØØØ
44.       1         37.Ø     Ø.Ø         7.    -199.    -35Ø.    1.ØØØ

45.  48       72Ø.Ø     Ø.12      1.Ø    -125.    -169.    26.51    1.ØØØ
46.       1         3Ø.Ø     Ø.Ø        -3.    -199.    -35Ø.    1.ØØØ

47.  DO YOU WANT TO SAVE DATA?  NO

48.  IS THERE ANOTHER ENVIRONMENT?  NO

49.  IS THERE ANOTHER SHIP/TARGET SET?  NO

50.  IS THERE ANOTHER SET OF RADAR PARAMETERS?  NO

51.  PROGRAM FINISHED.

52.  STOP
```

Example 1 is the printed output from a PDP DEC-10 system time-sharing terminal for a sample run of SURSEM. Line numbers have been added at the left of the listing for reference, and user responses have been underlined. Immediately on initiating execution, SURSEM prints a line of identification (line 1) and asks the user if he needs a description of the input variables (line 2). The user types either YES or NO. A YES answer will produce a descriptive listing of the radar, target, environment, and output control parameters discussed earlier. A NO answer, as shown here, causes the program to immediately print line 3. Detailed output consists of a line of 15 output values for each radar scan performed, modified by the output control parameters specified in line 10. A NO answer results in no output values printed at the terminal. Line 4 instructs the user to enter the 11 basic radar input parameters described earlier in the section on radar input. In this example the parameter values as entered are separated by spaces. Line 5 requests the 16 input parameters to define scan mode 1. If more than one scan mode is designated as the tenth basic input parameter in line 4, the program requests inputs for scan mode 2 after receiving the 16 entries in line 5 for scan mode 1 and continues requesting scan mode inputs until all scan modes are defined. Here only a single scan mode is to be defined, after which the program continues to line 6, an instruction for the user to enter the radar position coordinates. Line 7 requests the number of targets to be defined. Line 8, in reply, instructs the user to enter a set of 13 input target parameters for each target designated in line 7, one set at a time. Once all target information has been entered (only one set in this example), the program requests the environmental parameters to be entered in line 9. The last input information, the output control parameters, is requested in line 10. Once these six values have been entered, the scenario has been defined and the engagement begins.

Prior to the occurrence of the initial selected radar scan mode, the output heading, lines 11 and 12, is printed. Thereafter a line of output values is printed out at the terminal after each radar scan occurrence, subject to the conditions defined by the output control parameters. If no detailed output is desired, indicated by a NO answer to line 3, both the printing of the output heading and scan output values are omitted. At the conclusion of the engagement, line 47 is printed, offering the user the option to save the computer output data (from *all* scans actually performed for which the single-scan probability of detection equals or exceeds 0.001) on magnetic tape or disc file. (See discussion of the SURSEM Executive Routine and program listing in Appendix A regarding specification of, and writing on, magnetic tape or disc file for data storage.) Lines 48 through 50 offer the user the opportunity to perform another engagement using various parts of the previously defined scenario, thereby limiting the amount of additional input required. A response of YES to line 48 results in the program cycling back to line 9 and continuing on for another engagement from that point. Thus the scenario would retain the radar and target parameters previously defined and require only environmental and output control parameters (including scan modes to be used) to be entered. An answer of NO to line 48 and YES to line 49 results in a transfer to line 6 and a request for new radar position parameters. In this case only the basic radar and scan mode inputs from the preceding scenario are retained. Then the program continues through lines 7, 8, 9, 10, etc., as before. A response of NO to lines 48 and 49 and a response of YES to line 50 transfer the user to line 1 at the beginning of the program to define a completely new scenario. A NO answer to line 50 terminates the program. Note the first YES answer to lines 48 through 50 results in a transfer to the corresponding location in the input sequence, and all parts of the scenario defined after that point in the input sequence must be redefined.

In Example 1, the output index numbers corresponding to lines 13 through 20 are all 1. Each scan performed must result in a single-scan probability of detection of at least .001 before it is stored for possible transfer to magnetic tape or disc file at the conclusion of the engagement. Currently, provision is made to store a table of up to 4000 such values during the course of the engagement. The index printed is actually the index used to create the table, which starts with an index of 2, and its value is stepped up only when a detection value meets the .001 criterion and is stored. However, the results of all scans performed are printed (subject to output control parameter specifications). In this example, the first four full lines of output, all with index 1, have corresponding single-scan probabilities of detection less than .001 and thus were printed but not stored. Lines 21 and 22, with an index of 2, have a corresponding single-scan probability of detection of .002, and at this point the output control specification to *print only every fourth line of possible output* comes into effect, along with storage of the detection data, so that the next scan printed, lines 23 and 24, has an index of 6.

2.4 Batch Processing

The scenario input is accomplished in a batch-processing operation through the use of data input cards that follow the program card deck. The information required is the same as in the time-sharing version, and the same options are available. A SURSEM FORTRAN program card deck suitable for batch processing exists at NRL. The data cards required are described below.

Data card 1—Detailed output control integer (I5 format); 1 = detailed output to be printed

Data card 2—Title card, run identification (free-form)

Data card 3—Radar position (x, y, z) in thousands of feet (3F8.2 format)

Data card 4—Eleven basic radar parameters (9F8.2,I2,F6.2 format):

1. Radar frequency in megahertz
2. Antenna pattern function indicator 0 = pencil beam, 1 = cosecant-squared beam
3. Receiver noise in dB
4. Horizontal 3-dB beamwidth in degrees
5. Vertical 3-dB beamwidth in degrees
6. One-way antenna gain in dB
7. One-way sidelobe level in dB down from peak
8. Receiver loss in dB
9. Transmitter loss in dB
10. Number of scan modes
11. Linear polarization in degrees (0° = horizontal, 90° = vertical)

Data cards 5 and 6 (one set for each radar scan mode)—Sixteen parameters for each scan mode (10F8.2/6F8.2 format):

1. Lower boundary elevation-angle coverage in degrees
2. Upper boundary elevation-angle coverage in degrees
3. Peak power in megawatts
4. Pulse length in microseconds
5. Interlook period in seconds
6. Scan offset in seconds
7. Instrumented range in nautical miles
8. Mode-defendent loss in dB
9. Number of pulses integrated
10. Minus $\log_{10}$ (false-alarm probability)
11. Compressed-pulse length in microseconds
12. Sea-clutter improvement factor in dB
13. Intermediate-frequency bandwidth in megahertz; if 0, bandwidth will be set at 1.0/compressed-pulse length
14. Mode-dependent frequency increment in megahertz
15. Blanking time in microseconds; if 0, set at pulse length
16. Rain-clutter improvement factor in dB

Data card 7—Number of targets (I5 format)

Data card 8—Thirteen parameters for each target (12F6,2,I3 format):

1-4. Initial coordinates x, y, z in thousands of feet and time in seconds

5-8. Terminal coordinates x, y, z in thousands of feet and time in seconds

9-11. Radar reflective areas for head-on, broadside, and minimum in square meters

12. Jamming power density in watts per megahertz

13. Marcum-Swerling cross-section model number

Data card 9—Four environmental parameters (4F8,2 format):

1. Wind speed in knots
2. Height of wind-speed measurement in thousands of feet
3. Multipath indicator (1 = multipath, 0 = no multipath)
4. Rainfall rate in millimeters per hour

Data card 10—Print control parameters (3F8.2,10I4 format):

1. Lower bound for which single-scan probability of detection will be listed
2. Upper bound for which single-scan probability of detection will be listed
3. Largest cumulative probability of detection to be listed
4. Line print indicator; if this number were 5, then one out of every five lines of possible output would be listed
5. Number of modes to be used
6. Mode numbers to be used

Data card 11—Recycle control parameter (I5 format):

1 = New radar, in which case the next data card is card 1

2 = New targets

3 = New environment

4 = Run completed

3 SURSEM PROGRAM

3.1 SURSEM Executive Routine

The SURSEM Executive Routine drives the surveillance radar systems evaluation model. The model itself consists of a number of subroutines that perform specialized functions; the executive routine links these routines together. The modular construction of the model facilitates changes and additions to the existing version.

The executive routine initiates communication with the user, directly by means of a keyboard terminal in a time-sharing system or through data cards in a batch-processing operation. It monitors the input of scenario data (using Subroutines INITIAL, TARGETS, and ENVIRN) and print control parameters (read directly). On the time-sharing system, the user is given the option of having a description of the scenario parameters printed prior to request for input. The executive routine also performs the housekeeping chores for the model; it sets constants and conversion factors, and maintains various counters and flags to keep track of the progress of the engagement.

Once the scenario has been defined, the executive routine calls on Subroutine MATCH to determine the time at which each radar scan mode will first illuminate the target. At this point program initialization is complete and the executive routine enters its recursive portion as the engagement begins. A call to Subroutine NEXTSCN determines the scan mode that will next illuminate the target. Current engagement time now becomes the time of occurrence of the selected scan mode. Subroutine NEWPOST computes the current position of each target in preparation for the calculation of the single-scan probability of detection, which follows in Subroutine DETPROB. The cumulative probability of detection (the probability that the target has been detected at least once by this time into the engagement) is then calculated within the executive routine, which also stores current time and single-scan probability of detection in a detection table for optional future use.

If detailed output is desired and the conditions of the output control parameters are met, the output data corresponding to the scan mode just processed are scaled and printed. If the target has not yet reached its final destination, Subroutine NEXTSCN is called to determine the scan mode that will next illuminate the target, and the sequence of events described above is repeated.

The engagement ends when the target arrives at its terminal position. At this time the user is presented with several options. He may specify that the detection table from the completed engagement be saved on disk file or magnetic tape. He may conclude the run or he may elect to repeat the engagement, changing some or all of the scenario input parameters. The latter feature is particularly effective on the time-sharing system, in which the executive routine queries the user as to which input parameters to change; this results in a considerable saving of time and effort when only a limited number of parameter changes are involved.

The dictionary of variables for the SURSEM Executive Routine is given in Table 3.1-1. The SURSEM executive macro and Fortran flowcharts are shown in Figs. 3.1-2 and 3.1-3, respectively.

Table 3.1-1—SURSEM Executive Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ACON		Constant used in sea-state calculations (see Subroutine JAM)
ALPHAD		Grazing angle of clutter patch (deg)
AMBN	N_T	Thermal noise energy (J)
ANS1		Detailed output control; if YES, detailed output to be printed; if NO, no printed output
ANS2		Recycle control parameter; if YES new case with new ship/target set or new case with new radar parameters, depending on query; if NO, continue query
ANS3		Input parameter description control; if YES, detailed description of input parameters will be printed; if NO, description skipped
BETA		Constant used in sea-state calculations (see Subroutine JAM)
BVDEG	β_v	Target elevation measured from the horizon (deg)
BHDEG	β_h	Target azimuth (deg)
CCM		Speed of light (cm/s)
CNM		Speed of light (n.mi./s)
CONV	$10 \log_{10} e$	Conversion factor for converting natural logarithms to dB
CUM (I, J)		Array used to store single-scan probability of detection ($J = 2$) and time ($J = 1$) of the Ith detection event (for subsequent use by other programs)
CUMP		Largest cumulative probability of detection that will be listed
DBE		Signal energy (dB relative to a joule)
DBN		Total noise energy (dB relative to a joule)
DWL (J)		Frequency increment for mode J (Hz)
ENDTIME		Game time limit (hr) (set in Subroutine ENVIRN to be slightly greater than time target reaches terminal position)
FAC4	F	Propagation factor to the fourth power
FOPIQB	$(4\pi)^3$	Constant
FOPISQ	$(4\pi)^2$	Constant
GN		One-way antenna gain (basic radar parameter)
IANS		Recycle control parameter; if YES, new case with new environment; if NO, continue query

Table 3.1-1—SURSEM Executive Variable Dictionary (Continued)

FORTRAN Variable	Algebraic Notation	Description
IKEYF		Indicator 0 = program considering the event that is the first look at target by a mode 1 = program considering subsequent event
IREP		Print control parameter; e.g., IREP=5 indicates that every 5th line of output will be printed
IREPF		Counter for the number of events that are within program print control limits
ISTEP		Counter for the number of detection events (limited to 4000)
MC		Number of scan modes operational for engagement
MCC(I)		Scan mode number of *I*th operational scan mode
MUL		Multipath indicator (1 = multipath, 0 = no multipath)
NSCAN		Number of scan modes defined during radar input
OFR		Frequency of mode under consideration (MHz)
PDCUM		Cumulative probability of detection
PDSS		Single-scan probability of detection
PBBS		Elevation of pencil-beam boresight for mode under consideration (rad)
PS1		Lower bound for which single-scan probability will be printed
PS2		Upper bound for which single-scan probability will be printed
RC(1)		Basic radar frequency (MHz)
RC(4)		Horizontal 3-dB beamwidth for mode under consideration (degrees to radians)
RC(5)		Vertical 3-dB beamwidth for mode under consideration (degrees to radians)
RC(6)		One-way antenna gain for mode under consideration
RC(12)		Signal energy (*J*)
RC(14)		Total noise energy (*J*)
RE		4/3 Earth's radius (m)
RFR		Ratio of basic frequency to frequency of mode under consideration
RMODE(*J*,5)		Lower elevation limit for mode *J* (rad)
SCDB		Total sea-clutter energy (*J*)

Table 3.1-1—SURSEM Executive Variable Dictionary (Continued)

FORTRAN Variable	Algebraic Notation	Description
T		Game time (hr)
THH		Horizontal 3-dB beamwidth for basic mode (rad)
THV		Vertical 3-dB beamwidth for basic mode (rad)
TRGP		Slant range of target 1 (n.mi.)
TRGPOS(1,4)	R	Slant range of target 1 (n.mi.)
TRGPOS(1,5)		Target azimuth (rad)
TRGPOS(1,6)		Target elevation measured from horizon (rad)
TSEC		Game time (s)
V		Range extent of clutter cell (m)
XJAMN		Total jamming and sea-clutter energy (J)
XJAMNDB	E_J	Total jamming and sea-clutter energy (dB relative to a joule)
XKTOMS		Conversion factor (knots to m/s)
XNMTOM		Conversion factor (nautical miles to meters)

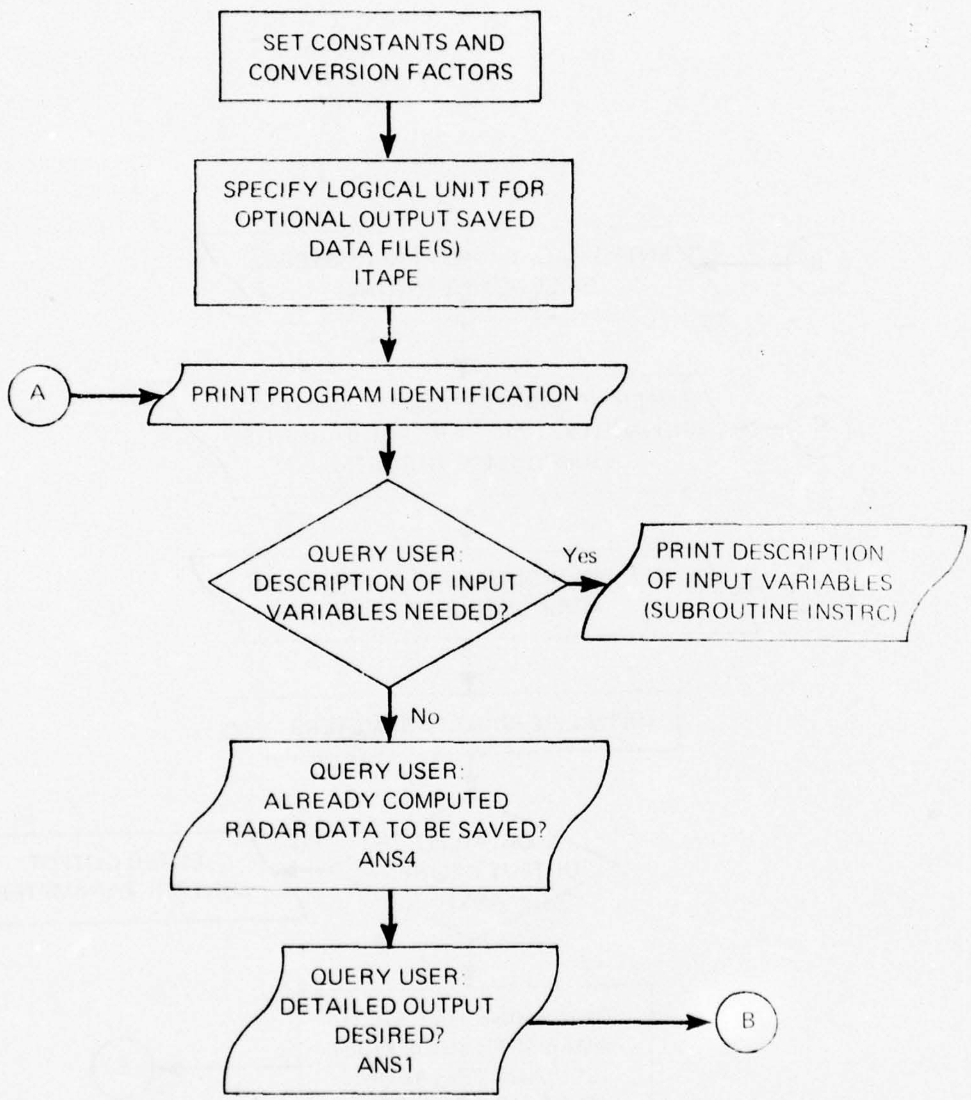


Fig. 3.1-2—SURSEM executive macro flowchart

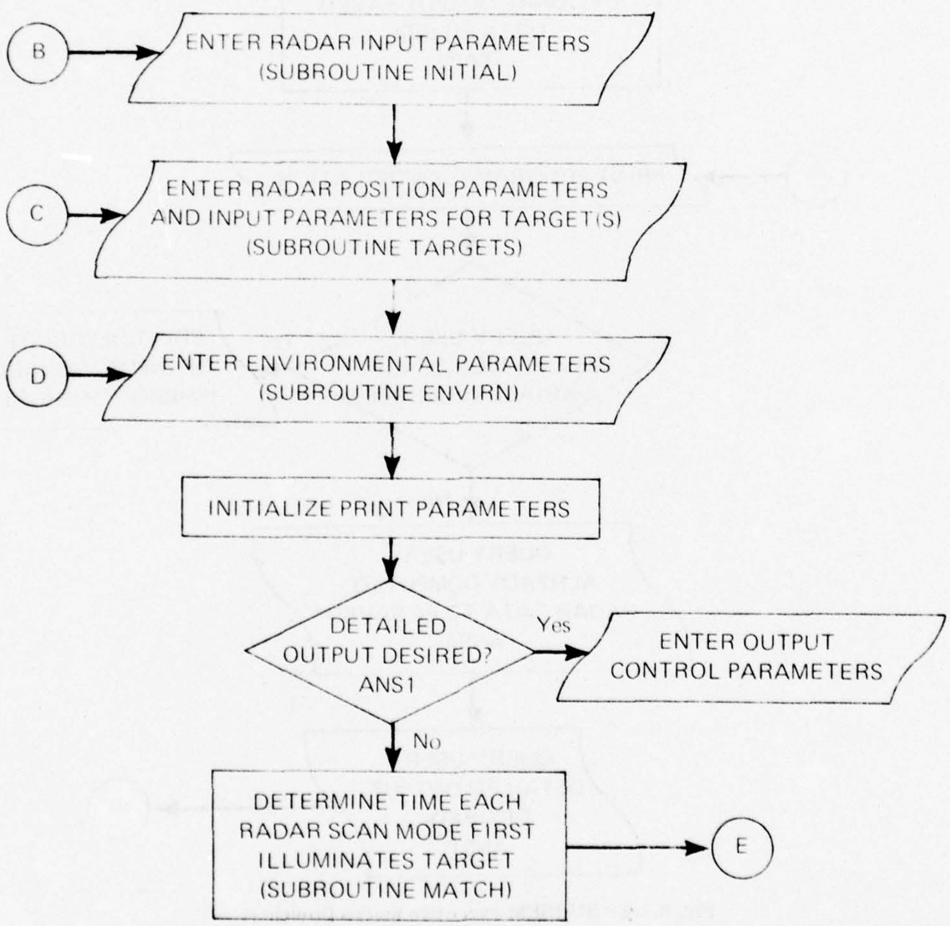


Fig. 3.1-2—SURSEM executive macro flowchart (Continued)

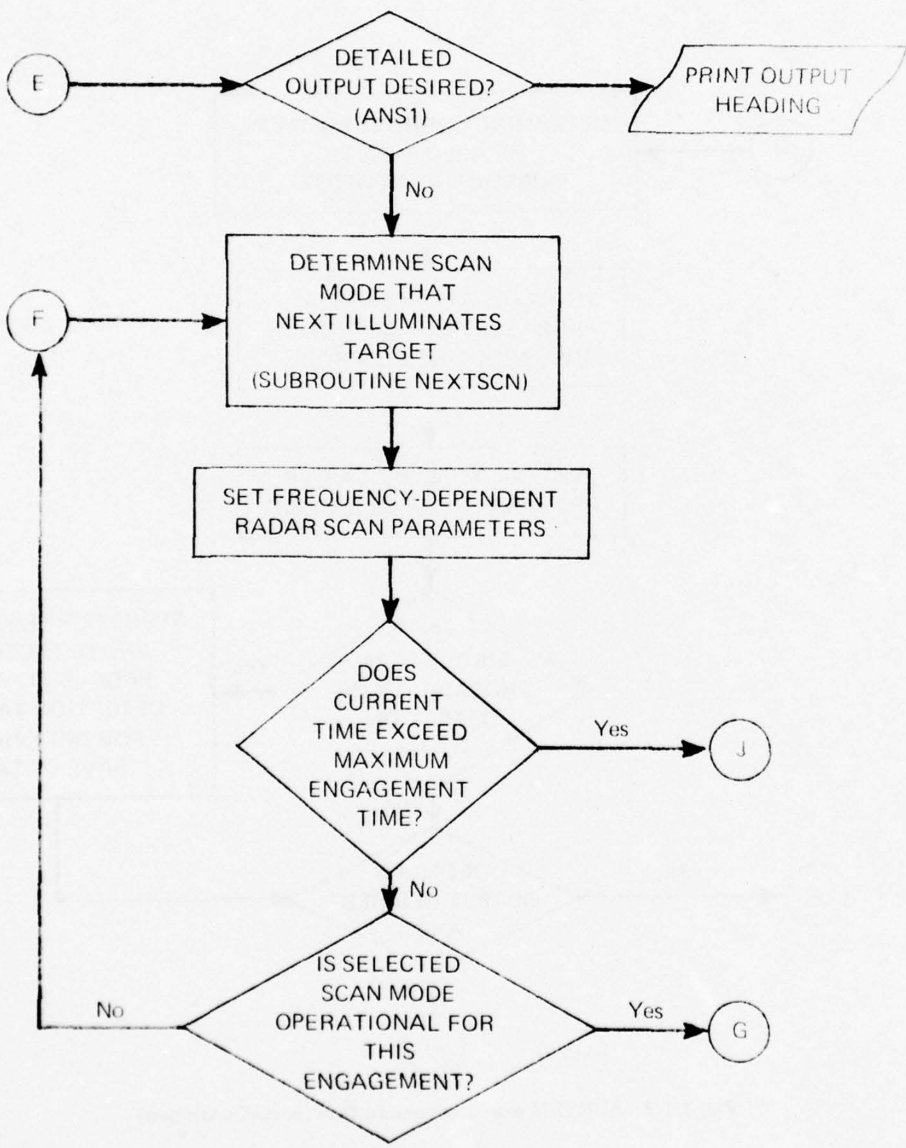


Fig. 3.1-2—SURSEM executive macro flowchart (Continued)

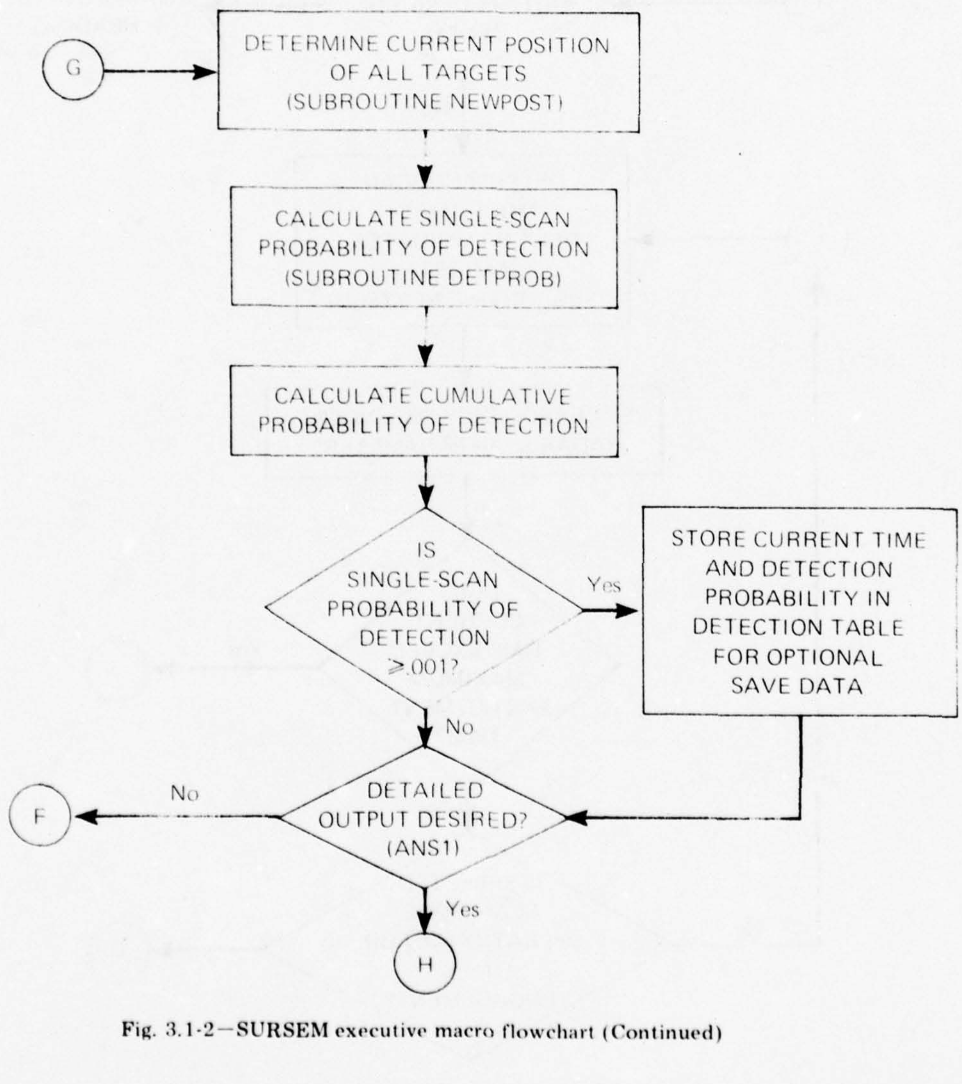


Fig. 3.1-2—SURSEM executive macro flowchart (Continued)

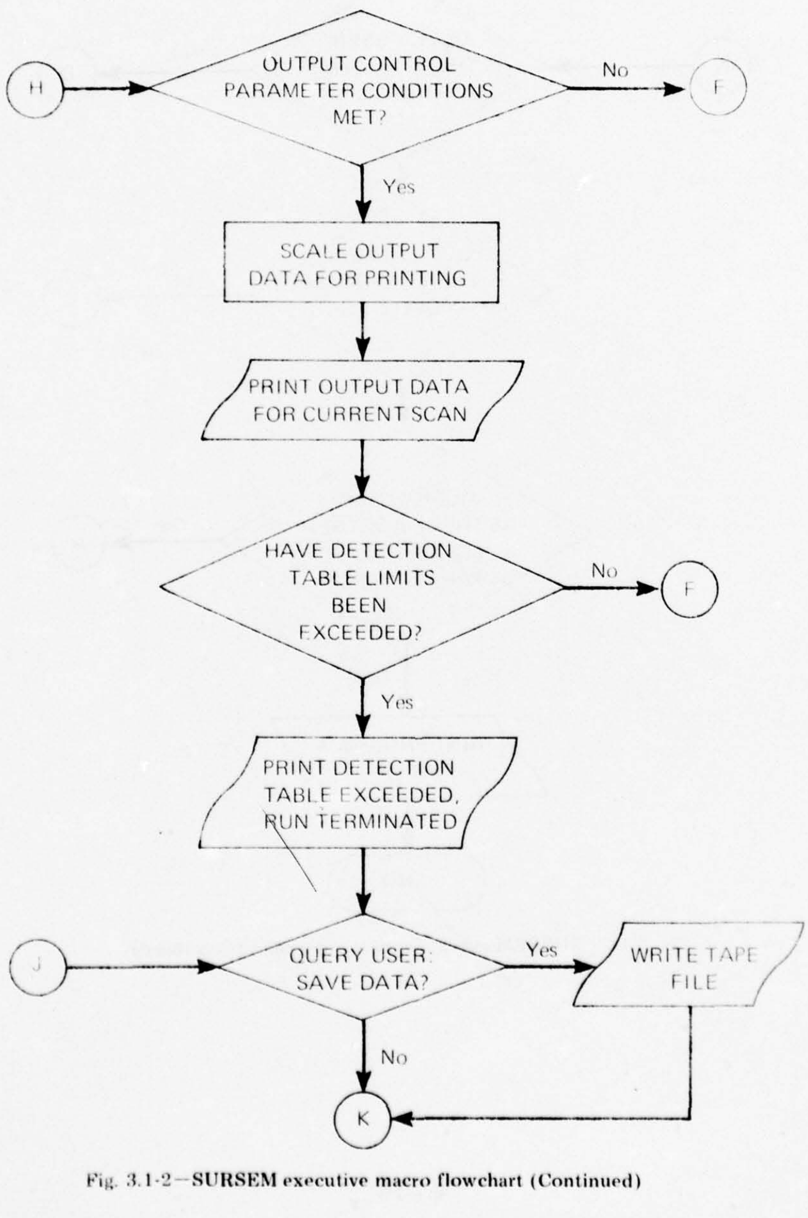


Fig. 3.1-2—SURSEM executive macro flowchart (Continued)

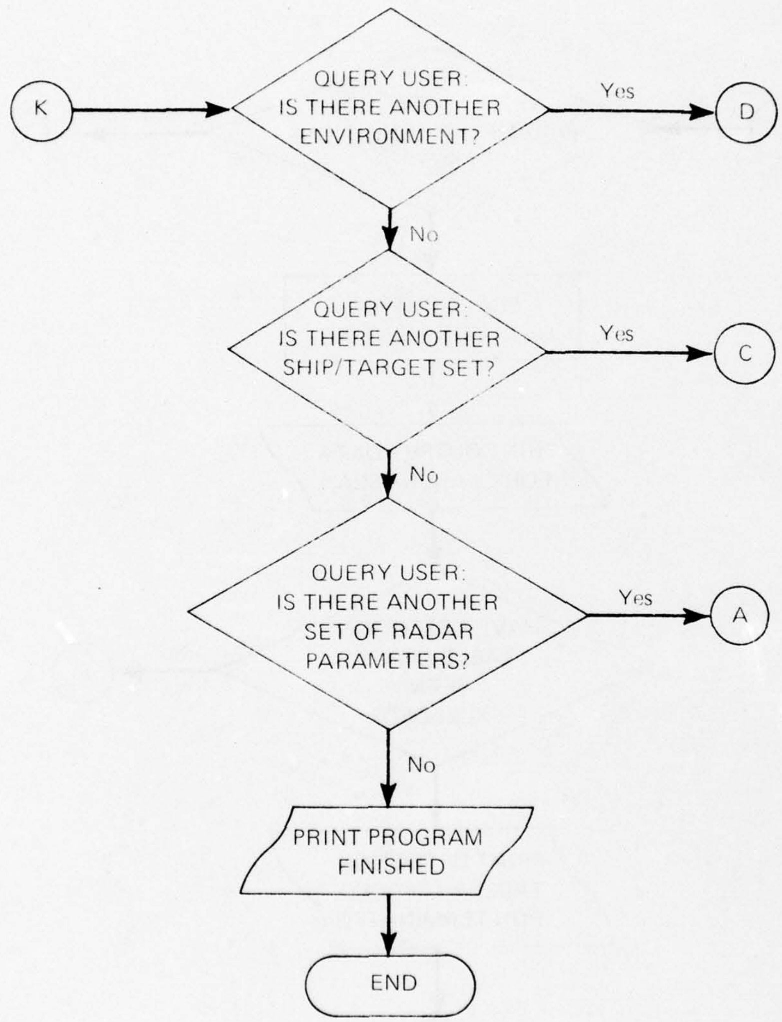


Fig. 3.1-2—SURSEM executive macro flowchart (Continued)

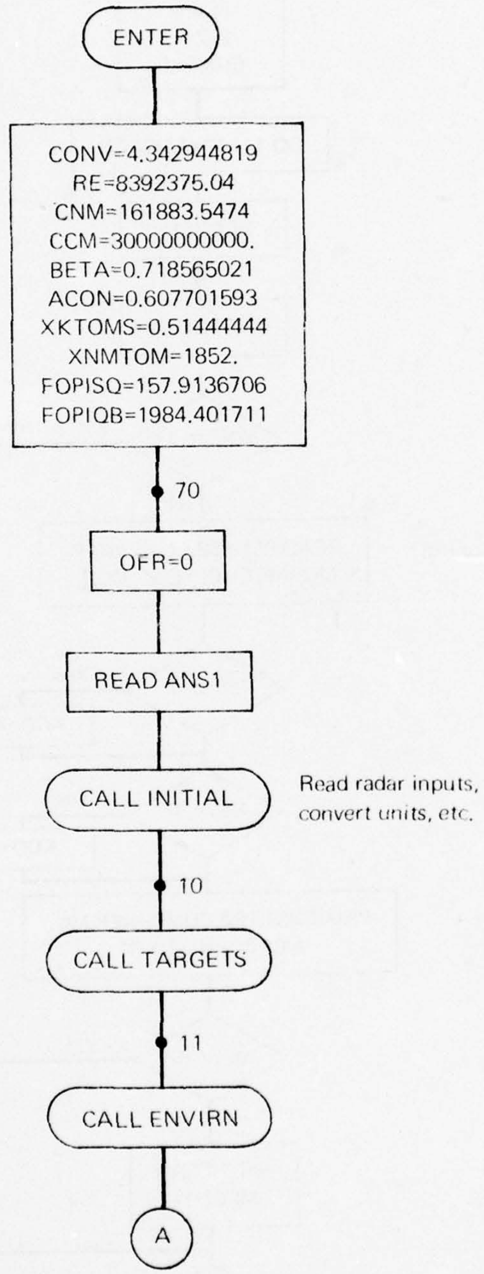


Fig. 3.1-3—SURSEM executive flowchart

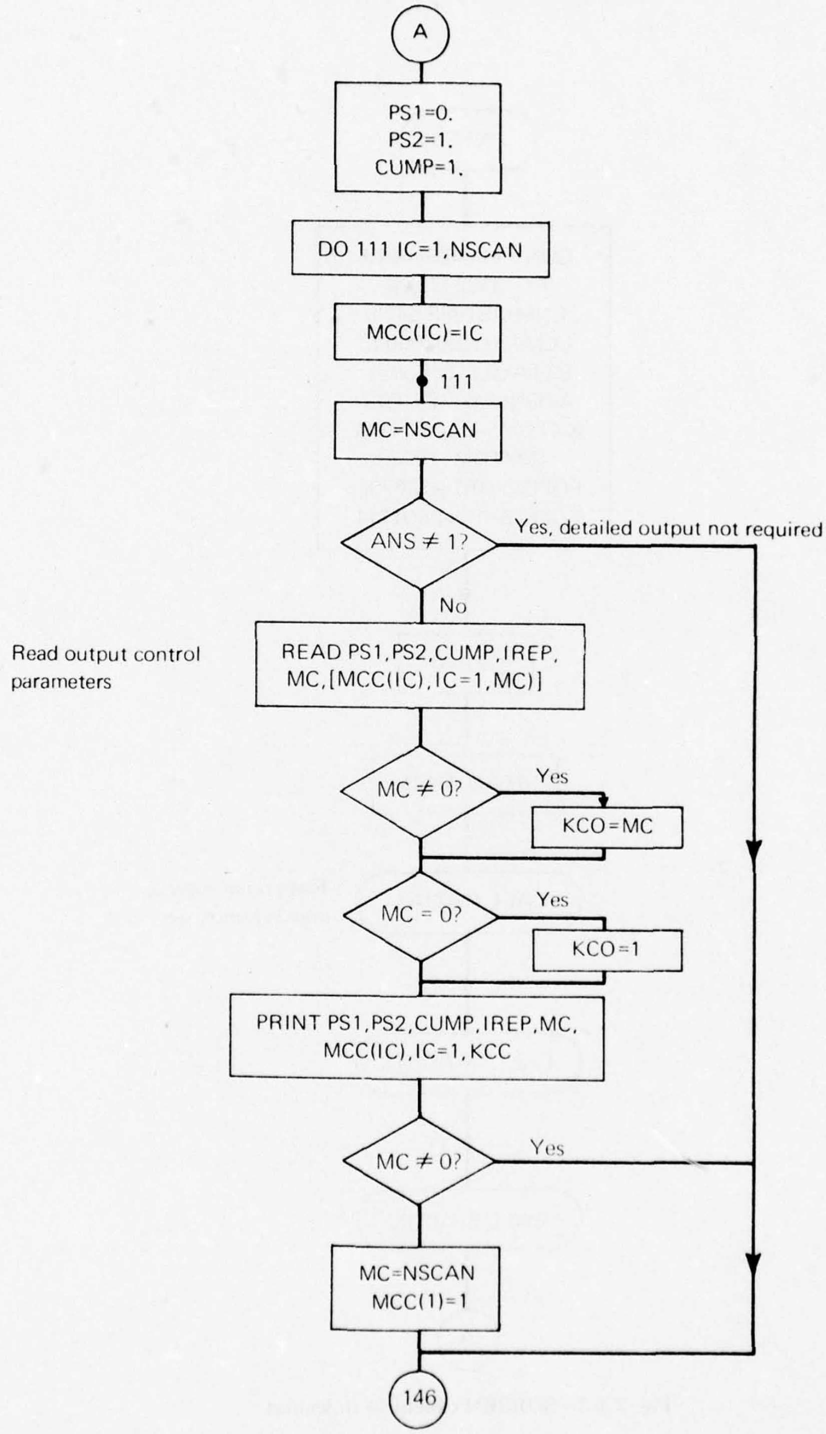


Fig. 3.1-3—SURSEM executive flowchart (Continued)

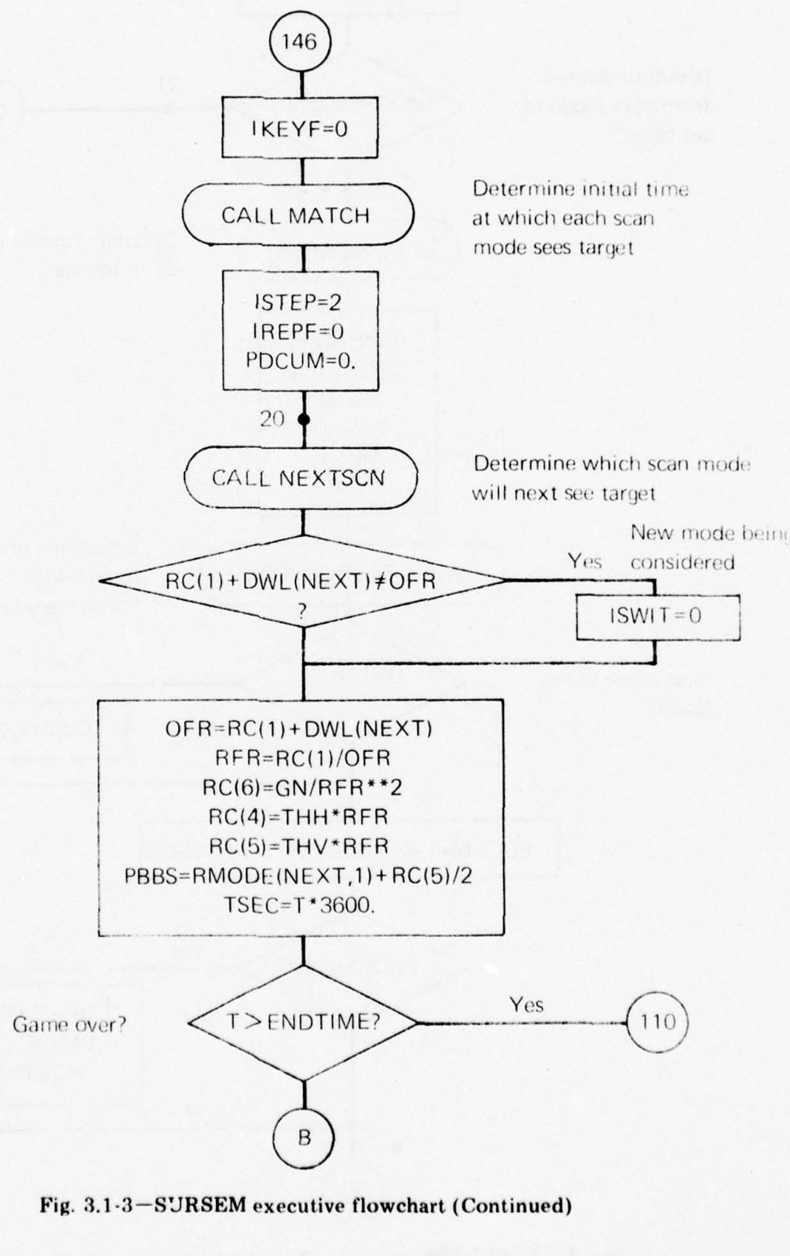


Fig. 3.1-3—SURSEM executive flowchart (Continued)

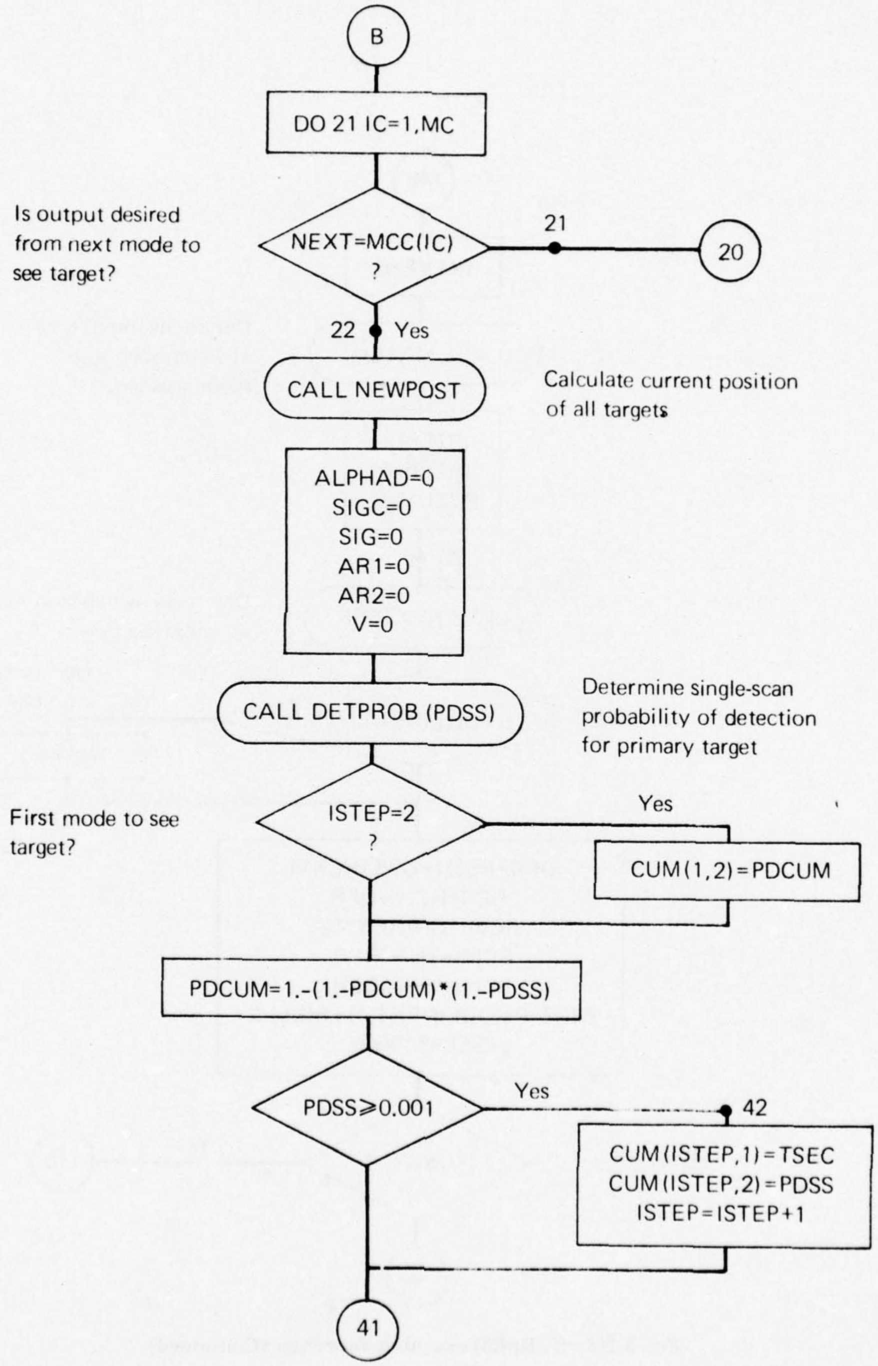


Fig. 3.1-3—SURSEM executive flowchart (Continued)

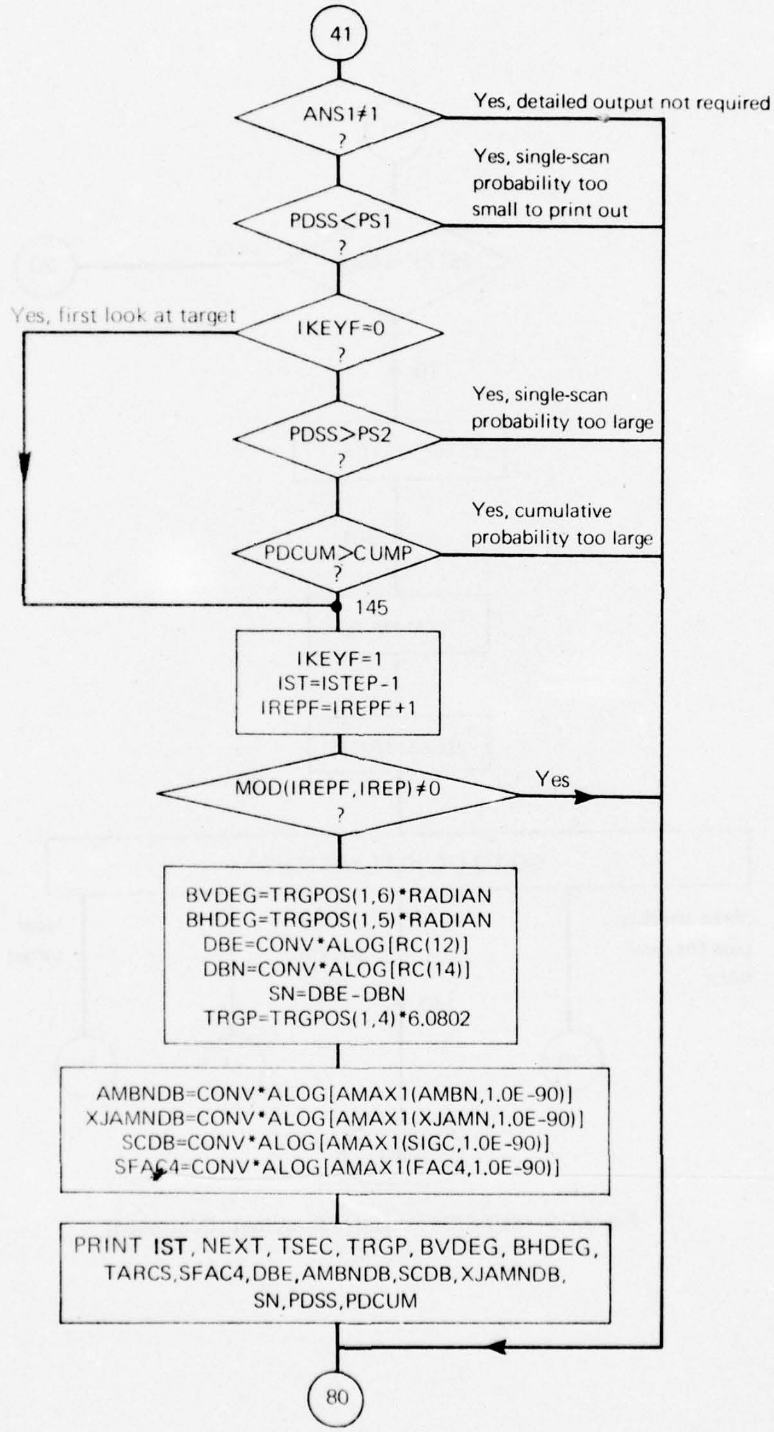


Fig. 3.1-3—SURSEM executive flowchart (Continued)

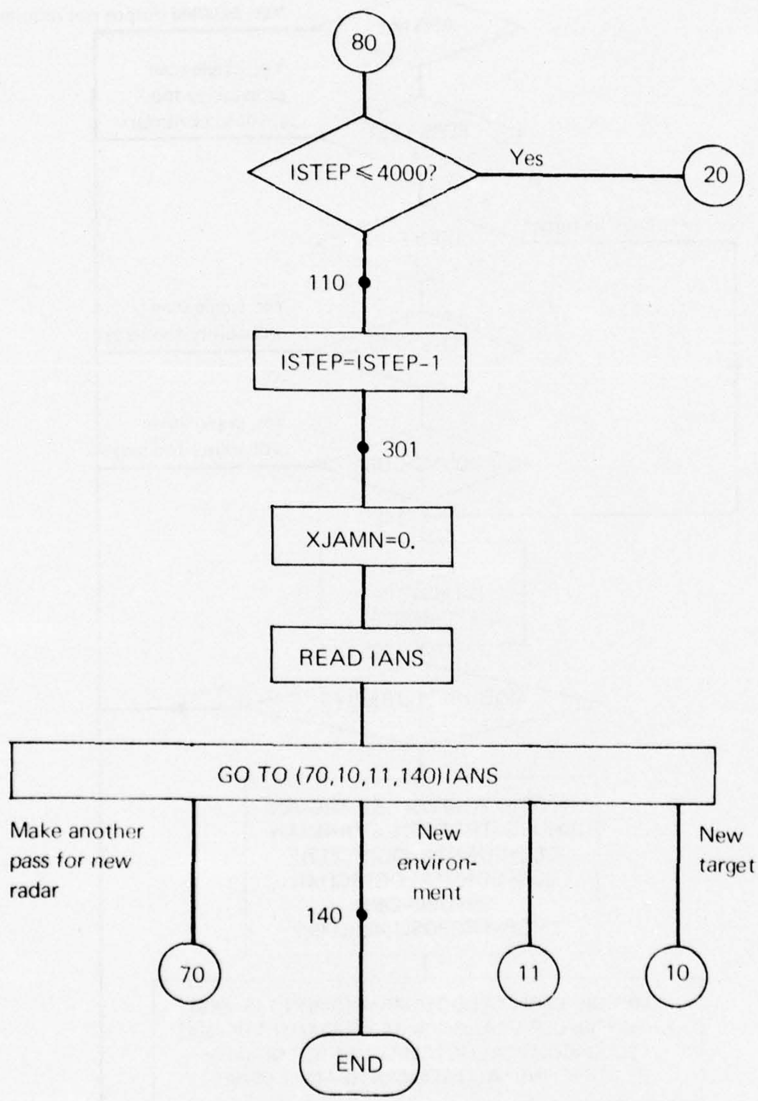


Fig. 3.1-3 -SURSEM executive flowchart (Continued)

3.2 Function BEAM

Function BEAM is used by the JAM, DETPROB, and GAIN subroutines to determine the normalized beam pattern factor. The function is capable of handling cosecant-squared and pencil-beam patterns. The dictionary of variables and the BEAM flowchart are presented in Table 3.2-1 and Fig. 3.2-2, respectively.

The function is called with four calling parameters: ALPHA, BETA, GAMMA, and KEY1. ALPHA is the angle between the pencil-beam boresight and target, measured positively in the clockwise direction (see Fig. 3.2-1). For cosecant-squared beams ALPHA is measured from the center of the main beam. BETA is the 3-dB beamwidth. GAMMA is an indicator that specifies whether the beam pattern factor is being determined in the horizontal or the vertical direction, and KEY1 is also an indicator that identifies the beam type.

The BEAM function uses a (sin x)/x curve to represent the horizontal and vertical beam patterns of the pencil beam and the horizontal beam pattern of the cosecant-squared beam. The vertical beam pattern of the cosecant-squared beam is modeled by a main beam with a (sin x)/x representation and an extended fan above the main beam in which the antenna gain will vary with elevation angle according to R^2/h^2.

Table 3.2-1 — BEAM Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ALPHA	α	Angle between boresight and target for (rad)
BETA	β	The 3-dB beamwidth (rad)
DBDOWN		First sidelobe power level ratio
GAMMA	γ	Beam pattern indicator (0 = pencil beam, 1 = $\csc^2$ beam)
HOFK	ϕ_h	Angle at which the slope of the (sin px)/px beam pattern equals zero (rad)
KEY1		Beam pattern factor indicator (0 = horizontal, 1 = vertical)
PBBS	E	The elevation of the "boresight" of the main beam portion of the $\csc^2$ beam (rad)
RMODE(J,2)	E_u	Upper elevation limit for mode J (rad)
SINC	f	Normalized beam pattern factor (power)
THETA	θ	Normalized angular position of target (rad)
THETBK	β_k	The 3-dB beamwidth of the main beam portion of the $\csc^2$ beam multiplied by a constant k that sets THETBK equal to twice the elevation of the pencil-beam boresight (rad)

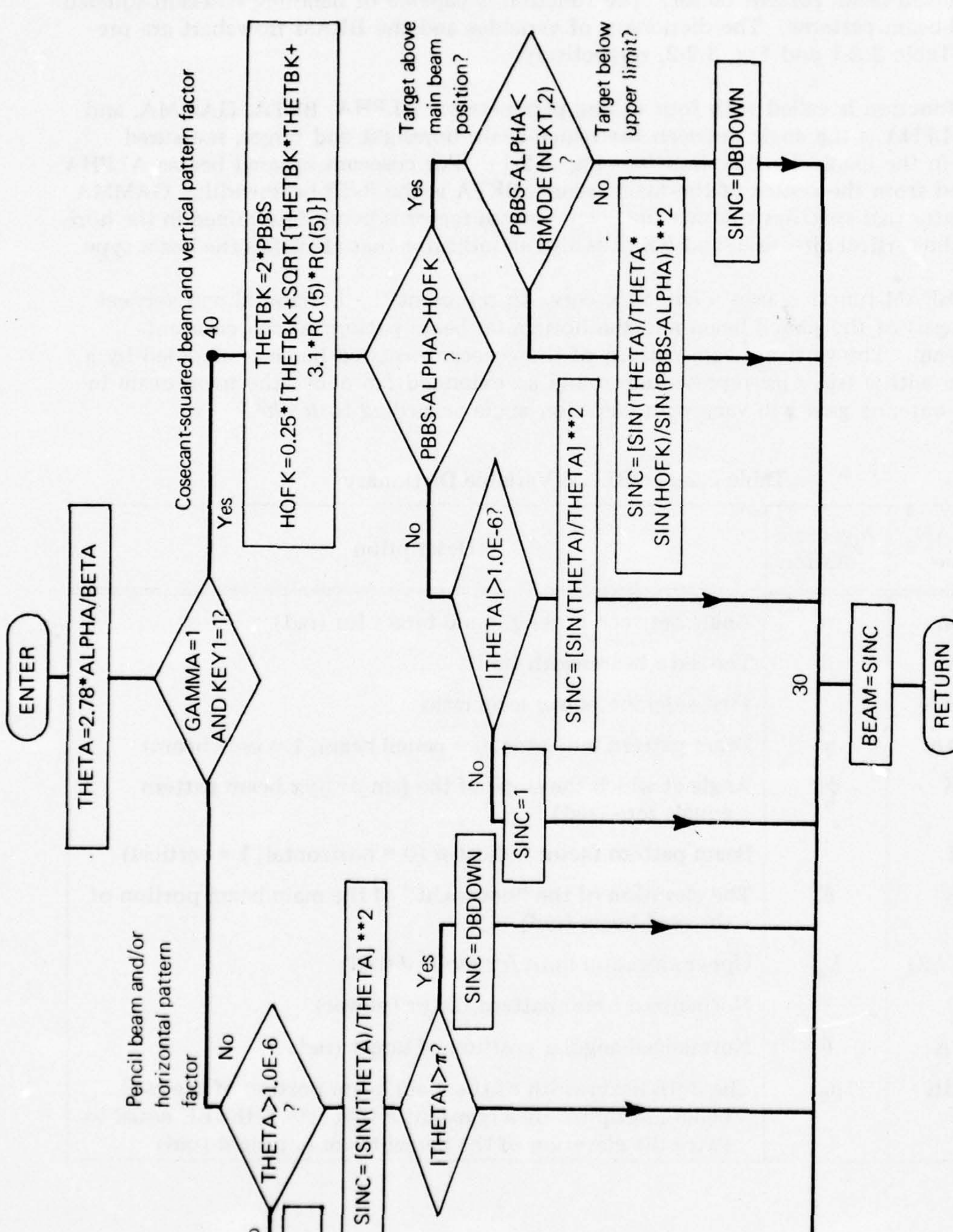


Fig. 3.2-2—BEAM(ALPHA, BETA, GAMMA, KEY1) flowchart

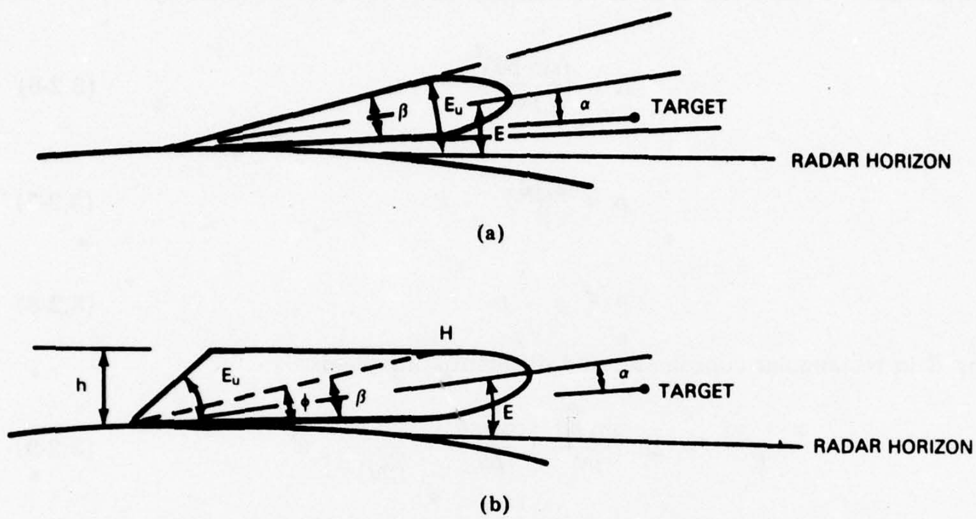


Fig. 3.2-3—Beam patterns: (a) pencil beam; (b) cosecant-squared beam

The first step in the calculation is to normalize the angle α; that is, the beam pattern is assumed to be a $(\sin k\alpha)/k\alpha$ curve with a 3-dB beamwidth given by β, so that the normalized angle θ on a $(\sin x)/x$ curve that corresponds to α is found from

$$\theta = \frac{2.78}{\beta}\alpha \tag{3.2-1}$$

where 2.78 is the 3-dB beamwidth on a $(\sin x)/x$ curve.

The next step in the calculation process depends on beam type and orientation. If the beam is a pencil beam or if the horizontal pattern factor is being determined for a cosecant-squared beam, then the beam pattern factor (power) is determined from

$$f = \left(\frac{\sin\theta}{\theta}\right)^2 \tag{3.2-2}$$

unless $\theta < 10^{-6}$ or $|\theta| > \pi$, in which case f is set as follows:

$$f = 1.0 \qquad (\theta < 10^{-6}), \tag{3.2-3}$$

$$f = f_{SL} \qquad (|\theta| > \pi) \tag{3.2-4}$$

where f_{SL} is the input sidelobe level.

When the beam is a cosecant-squared beam and the vertical beam pattern factor is being determined, the process is more complicated. The main beam is pointed at some angle E above the horizon (see Fig. 3.2-3). This can be expressed as a function of the 3-dB beamwidth β and a constant k:

$$E = \frac{k\beta}{2}. \tag{3.2-5}$$

The location of the point on the main beam at which the tangent to the beam is zero determines the start of the cosecant-squared portion of the beam. Assuming that the main beam can be represented by a $(\sin kx)/kx$ curve, this point is found as follows: A point on the curve with elevation ϕ has a radius R or normalized power magnitude given by

$$R = \left(\frac{\sin p\theta}{p\theta}\right)^2 \tag{3.2-6}$$

where

$$p = \frac{2.78}{\beta} \tag{3.2-7}$$

and

$$\theta = \phi - E . \tag{3.2-8}$$

Expressing R in rectangular coordinates and differentiating yields

$$\frac{x + yy'}{R} = 2p \frac{\sin p\theta}{p\theta} \left[\frac{\cos p\theta}{p\theta} - \frac{\sin p\theta}{(p\theta)^2}\right] \theta' \tag{3.2-9}$$

where

$$\theta = \tan^{-1} \frac{y}{x} \tag{3.2-10}$$

and

$$\theta' = \frac{1}{1 + y^2/x^2} \left(\frac{y'}{x} - \frac{y}{x^2}\right), \tag{3.2-11}$$

or

$$\theta' = \frac{xy' - y}{R^2} \tag{3.2-12}$$

At the angle θ_h where the cosecant-squared beam pattern starts, $y' = 0$. Combining Eqs. (3.2-9) and (3.2-12) and inserting $y' = 0$ gives

$$\frac{x}{R} = 2p \frac{\sin p\theta_h}{p\theta_h} \left[\frac{\cos p\theta_h}{p\theta_h} - \frac{\sin p\theta_h}{(p\theta_h)^2}\right] \frac{y}{R^2}, \tag{3.2-13}$$

or

$$R = 2k \frac{\sin p\theta_h}{p\theta_h} \left[\frac{\cos p\theta_h}{p\theta_h} - \frac{\sin p\theta_h}{(p\theta_h)^2}\right] \tan \phi_h . \tag{3.2-14}$$

Substituting the expression for R found in Eq. (3.2-6) gives

$$0 = 1 \div 2p \left(\cot p\theta_h - \frac{1}{p\theta_h}\right) \tan \phi_h . \qquad (3.2\text{-}15)$$

It is reasonable to assume that ϕ_h and θ_h are small. With this assumption it can be shown that

$$\cot k\theta_h - \frac{1}{p\theta_h} \approx \frac{p\theta_h}{3} \qquad (3.2\text{-}16)$$

and

$$\tan p\phi_h \approx p\phi_h . \qquad (3.2\text{-}17)$$

This reduces Eq. (3.2-15) to

$$\frac{2p^2\phi_h\theta_h}{3} = 1 \qquad (3.2\text{-}18)$$

or

$$2p^2\phi_h(\phi_h - E) = 3 , \qquad (3.2\text{-}19)$$

which can be expressed as the quadratic form

$$\phi_h^2 - \phi_h E - \frac{3}{2p^2} = 0 \qquad (3.2\text{-}20)$$

with the root

$$\phi_h = \frac{E + (E^2 + 6/p^2)^{1/2}}{2} . \qquad (3.2\text{-}21)$$

Using Eqs. (3.2-5) and (3.2-7), this can be further expressed as a function of β:

$$\phi_h = \frac{1}{2}\left[\frac{k\beta}{2} + \left(\frac{k^2\beta^2}{4} + \frac{6\beta^2}{(2.78)^2}\right)^{1/2}\right] \qquad (3.2\text{-}22)$$

or

$$\phi_h = \frac{\beta}{4}\left[k + (k^2 + 3.1)^{1/2}\right]. \qquad (3.2\text{-}23)$$

The normalized beam pattern factor (power) for the vertical pattern of the cosecant-squared beam can now be determined according to the position of the target. If the target elevation is above ϕ_h, the antenna beam pattern factor will vary with elevation according to the square of the cosecant:

$$f(\phi) = \frac{K}{\csc^2 \phi}$$

where K is readily found to be

$$K = f(\phi_h) \csc^2 \phi_h \tag{3.2-24}$$

and

$$f(\phi) = f(\phi_h) \left(\frac{\csc^2 \phi_h}{\csc^2 \phi} \right). \tag{3.2-25}$$

If the target elevation is less than or equal to ϕ_h,

$$f(\phi) = \left[\frac{\sin p(\phi - E)}{p(\phi - E)} \right]^2 . \tag{3.2-26}$$

In summary, the beam pattern factor for the vertical pattern of the cosecant-squared beam can be expressed as follows:

$$f(\phi) = \begin{cases} \text{sinc}^2 \dfrac{2.78}{\beta[\phi - (k\beta/2)]} & (\phi \leqslant \phi_h) \\[2ex] \text{sinc}^2 \dfrac{2.78}{\beta[\phi_h - (k\beta/2)]} \dfrac{\csc^2 \phi_h}{\csc^2 \phi} & (E_u > \phi > \phi_h) \\[2ex] \text{first sidelobe level} & (\phi > E_v) , \end{cases}$$

where the term sinc $(x) \equiv \sin x/x$.

This completes the computation process, and control of the program is returned to the calling subroutine.

3.3 Subroutine CLUTSIG

Subroutine CLUTSIG is called by Subroutine JAM to evaluate the normalized reflectivity σ_0, which corresponds to given values of radio frequency (XFRE), Beaufort sea state (XBEAU), incident angle (XANG), and the angle of linear polarization. The dictionary of CLUTSIG variables is given in Table 3.3-1, and the flowchart is shown in Fig. 3.3-2.

Table 3.3-1 — CLUTSIG Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ANG		Linear polarization (rad)
INDEX(I,J)		The INDEX (3,2) array identifies the values in the XPAR array that straddle the values of RF, Beaufort scale, and incidence angle that are being considered
NDX(I)		Parameter bookkeeping array: NDX(1) = number of RF considered NDX(2) = number of Beaufort scales considered NDX(3) = number of incident angles considered
PAR(1)		Parameter 1 is assigned the value of the RF if it is within the boundary values 500 to 35 000 MHz; otherwise it is assigned the boundary value that is closest to the RF
PAR(2)		Similar to parameter 1 but for the Beaufort scale with a range of 1 to 6
PAR(3)		Similar to parameter 1 but for 10 times the incident angle with a range of 1° to 100°
POLRZ	θ_p	Linear polarization (deg) (0° = horizontal)
SIGN		Normalizing factor applied to TEMP
SIGOH(I,J,K)		Three-dimensional array relating normalized reflectivity to RF(I), Beaufort scale (J), and incident angle (K), for horizontal polarization
SIGOV(I,J,K)		Similar to SIGOH but for vertical polarization
SIGZ	σ_0	Normalized reflectivity (dB below a 1-m^2 target/m^2)
SIGZP		Normalized reflectivity associated with vertical polarization (dB below a 1-m^2 target/m^2)
TEMP		Weighting function applied to the value of the normalized reflectivity at each vertex of a "cube" in the SIGOV or SIGOH matrix; the cube surrounds the point defined by PAR(J)
XANG		Incident angle (rad)
XBEAU		Beaufort scale
XFRE	f	RF (MHz)

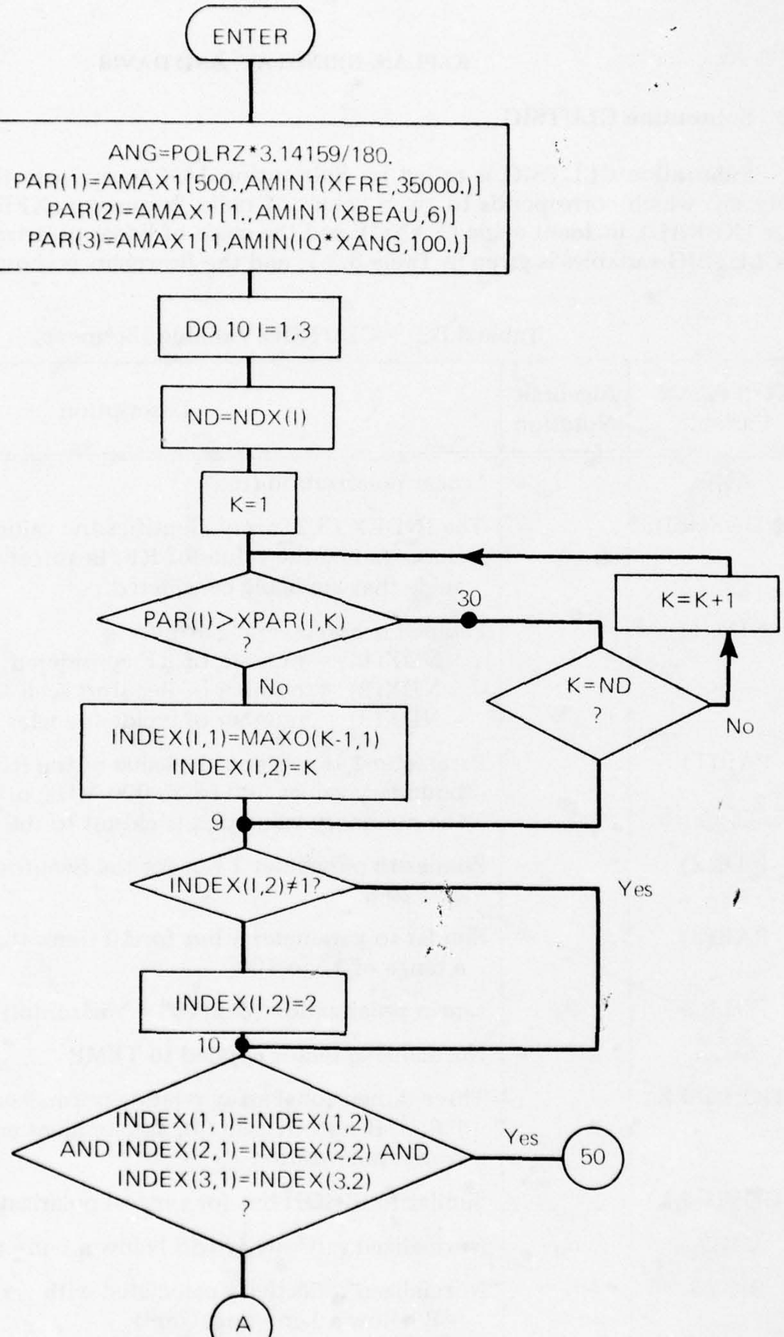


Fig. 3.3-2—CLUTSIG(XFRE, XBEAU, XANG, SIGZ, POLRZ) flowchart

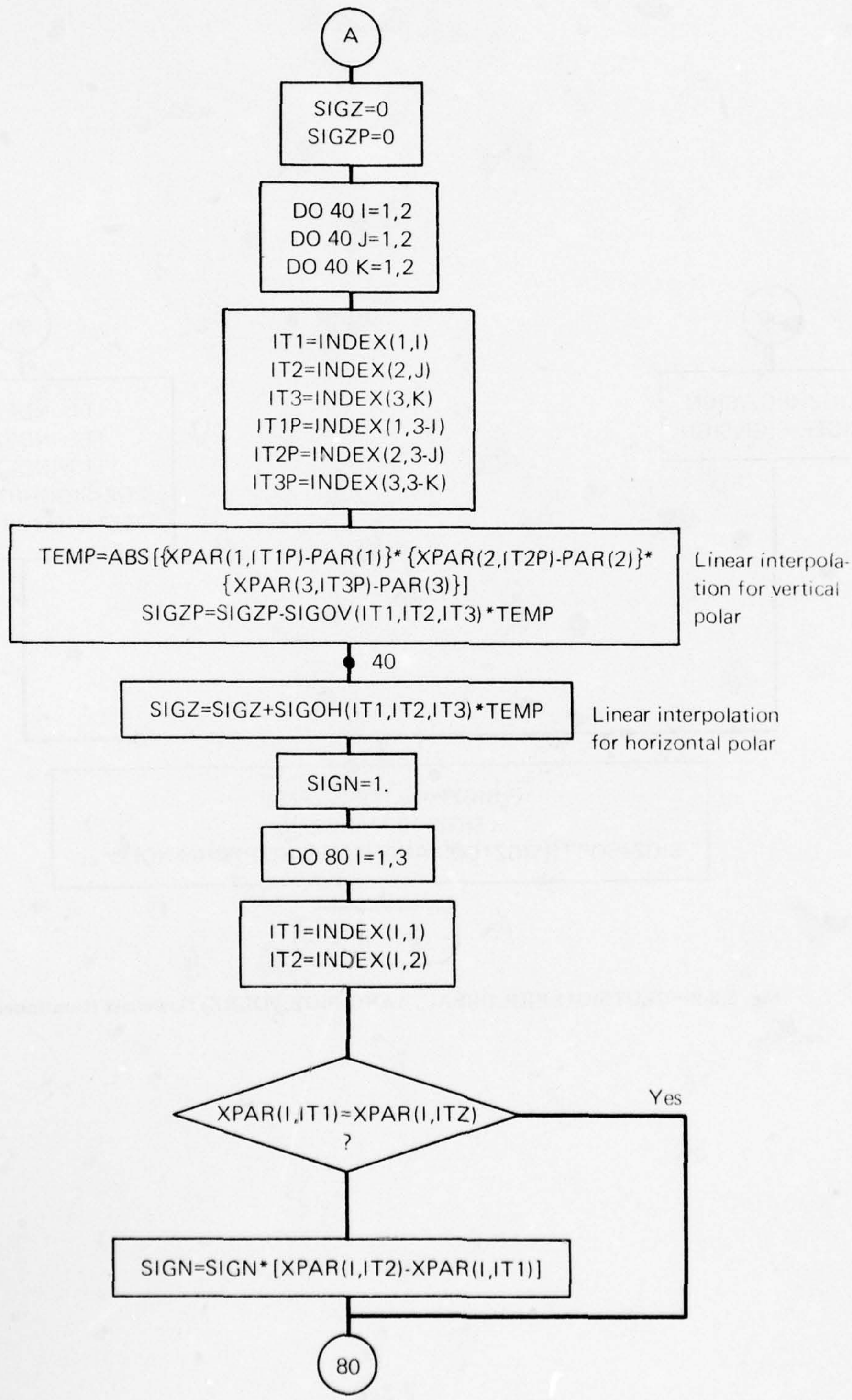


Fig. 3.3-2—CLUTSIG(XFRE, XBEAU, XANG, SIGZ, POLRZ) flowchart (Continued)

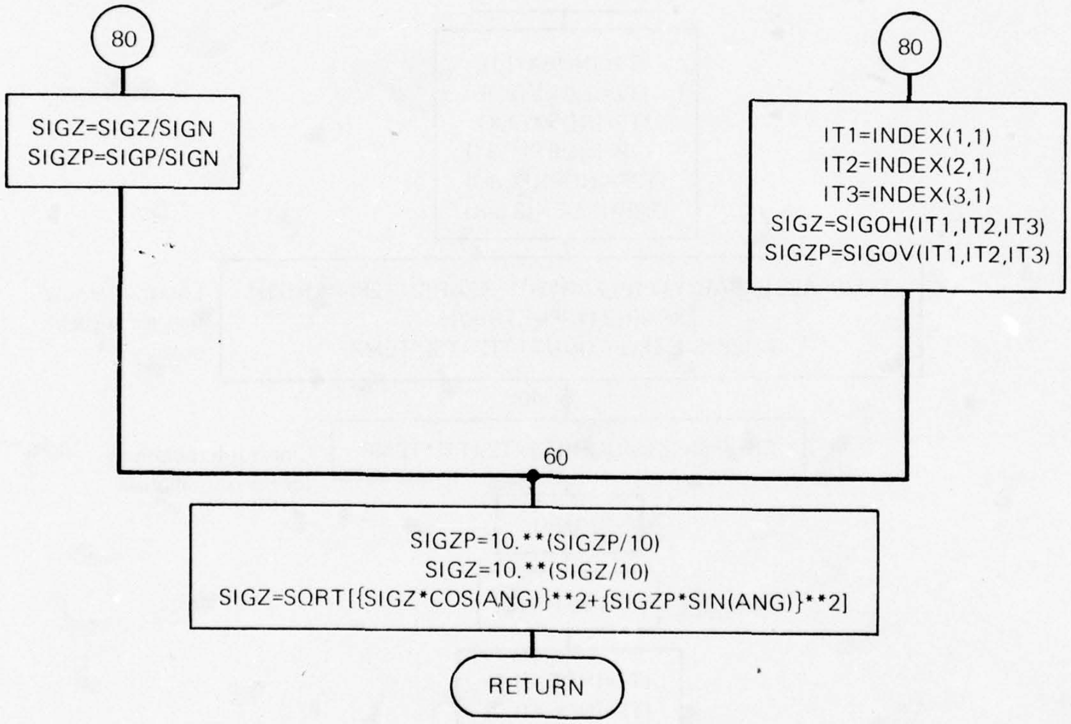


Fig. 3.3-2—CLUTSIG(XFRE, DBEAU, XANG, SIGZ, POLRZ) flowchart (Continued)

The normalized reflectivity represents the observed mean radar cross section from each unit of area in the clutter cell; that is, if $\sigma_0 = -20$ dB, each square meter in the clutter cell will contribute a radar cross section that is 20 dB below a 1-m^2 target.

Values of σ_0 for various radio frequencies (I), sea states (J), and incidence angles (K) are stored in two three-dimensional arrays: SIGOH(I,J,K) and SIGOV(I,J,K), corresponding to horizontal and vertical polarization, respectively. These values are based on tables that were presented in Ref. 2 and have been extended for greater utility. The range of values for the various parameters is as follows:

Parameter	*Range of Values*
Frequency, MHz	500, 1250, 3000, 5600, 9000, 17 000, 35 000
Beaufort scale	1, 2, 3, 4, 5, 6
Incident angle, degrees	0.1, 0.3, 1, 3, 10

In its current configuration, Subroutine CLUTSIG considers only linearly polarized radars. For a given set of parameters values of σ_0 are drawn from the SIGOH and SIGOV arrays by means of a linear interpolation scheme. These values of the normalized reflectivity are for horizontal and vertical polarization, respectively. The normalized reflectivity for a given linear polarization angle θ_p is found from

$$\sigma_0(\theta_p) = \left[(\sigma_{oH} \cos \theta_p)^2 + (\sigma_{oV} \sin \theta_p)^2\right]^{1/2} .$$

This value of σ_0 is returned to Subroutine JAM, thereby completing the process.

3.4 Subroutine DETPROB

Subroutine DETPROB is called by the SURSEM Executive Routine to determine the single-scan probability of detection for a given target position and environment. The process is detailed and takes into account such factors as target aspect angle, target fluctuations, multipath, rain attenuation, rain clutter, sea clutter, and jamming. The process is outlined below, followed by a flowchart of the subroutine (Fig. 3.4-1) and a list of FORTRAN variables (Table 3.4-2).

DETPROB first calls Subroutine TARSIG to compute the target's radar cross section σ as a function of aspect angle. This is then used to determine the free-space returning signal power at beam maximum according to

$$P_r = \frac{P_T G^2 \lambda^2 \sigma L_R L_T L_M}{R^4 (4\pi)^3} \tag{3.4-1}$$

where

P_T = peak power (watts)

G = one-way antenna gain (with respect to omnidirectional case)

λ = radar wavelength (meters)

σ = target cross section (square meters)

L_r = receiver antenna losses

L_T = losses between transmitter output and free space

L_M = mode-dependent losses

R = target range (meters)

Received signal energy for a pulse of duration τ is then given by

$$S_0 = P_r \tau . \tag{3.4-2}$$

If the radar under consideration is not a pencil beam or tracking radar, adjustments have to be made to the signal energy to account for the target's being off beam center. This is accomplished through the use of the BEAM function, which determines a one-way beam-shape factor $f(\theta)$ that is applied to the received-signal energy according to

$$S_1 = S_0 [f(\theta)]^2 . \tag{3.4-3}$$

Rain is modeled by a large number of independent scatterers, each of cross section σ_i and located within the radar resolution cell. This method is suggested by Skolnik [3]. Accordingly the total cross section is given by

$$\sigma_R = V_m \sum \sigma_i \tag{3.4-4}$$

where V_m is the volume of the radar resolution cell and

$$V_m = \left(\frac{\pi}{4}\right) R^2 \theta_B d_B \left(\frac{c\tau_c}{2}\right), \tag{3.4-5}$$

where θ_B and ϕ_B are the horizontal and vertical 3-dB beamwidths in radians, respectively, and τ_c is the compressed-pulse length in seconds.

It can be shown that if the raindrop diameter D is small in comparison to λ, then σ_i can be represented by

$$\sigma_i = \frac{\pi^5}{\lambda^4} |K|^2 D^6 \tag{3.4-6}$$

where K depends on the dielectric constant of the scatterer and K^2 is approximately 0.93 for water at 10°C when λ = 10 cm. Since the drop diameter is not a convenient parameter, expressions have been developed that relate drop diameter to rainfall rate (in millimeters per hour):

$$\sum_i D^6 = 200 r^{1.6}. \tag{3.4-7}$$

Combining Eqs. (3.4-4), (3.4-5), and (3.4-6) and using a value of 0.93 for K^2 gives the total rain cross section σ_R in the resolution cell as

$$\sigma_R = 6.706 \times 10^{-6}\ \tau_c \theta_B \phi_B R^2 r^{1.6} \lambda^{-4}. \tag{3.4-8}$$

This expression for σ_R is used to determine the rain's contribution to the total noise.

The two-way rain attenuation is calculated by fitting a curve to data published by Nathanson [4]. The data show a logarithmic relationship between attenuation and frequency that can be expressed algebraically as

$$2\alpha = 1.753 \times 10^{-3} f^{1.87} \tag{3.4-9}$$

where 2α is the two-way attenuation (decibels per nautical mile per millimeter per hour) and f is the frequency in gigahertz. The two-way attenuation A_r (absolute) for a given rainfall rate r, range R (meters), and wavelength λ (meters) is given by

$$\log_{10} A_r = -10^{-8} R r \lambda^{-1.87}. \tag{3.4-10}$$

Before the rain-attenuated signal energy can be computed, it is necessary to consider multipath effects. This is carried out in Subroutine MULPTH, which calculates the propagation factor F. The rain-attenuated signal energy for IF bandwidth B_{IF} is then

$$S = S_1 F^4 A_r [\min (B_{\text{IF}} \tau_c, 1)] \tag{3.4-11}$$

where the bandwidth adjustment is consistent with matched-filter analysis.

The four components that are considered to be contributory to noise energy—thermal noise, jamming energy, sea clutter, and rain backscatter—are now computed.

Thermal noise energy is determined from

$$N_T = F_n k T_0 \tag{3.4-12}$$

where F_n is the receiver noise figure, k is Boltzmann's constant, and T_0 is the system temperature in kelvins.

Jamming energy and sea clutter are calculated by Subroutine JAM.

Rain backscatter energy is computed according to

$$E_R = \frac{6\sigma_R P_r I_c \tau}{\sigma} \tag{3.4-13}$$

where I_c is the rain-clutter improvement factor and the factor 6 represents the effect of averaging returns for rain in the resolution cell over many multipath fade and reinforcement regions.

The ratio of signal energy to noise energy is now determined by

$$\frac{S}{N} = S[(N_T + E_j + E_c + E_R)\max(B_{\mathrm{IF}}\tau, 1)]^{-1} \tag{3.4-14}$$

where E_j and E_c represent the jamming energy and clutter energy, respectively, and the adjustment to the total noise energy allows for increased noise due to an unmatched IF bandwidth. The signal-to-noise ratio is finally utilized by Subroutine MARSWR, which provides the signal-scan probability of detection for a specified Marcum-Swerling cross-section model. This completes the process, and control of the program is returned to the main program.

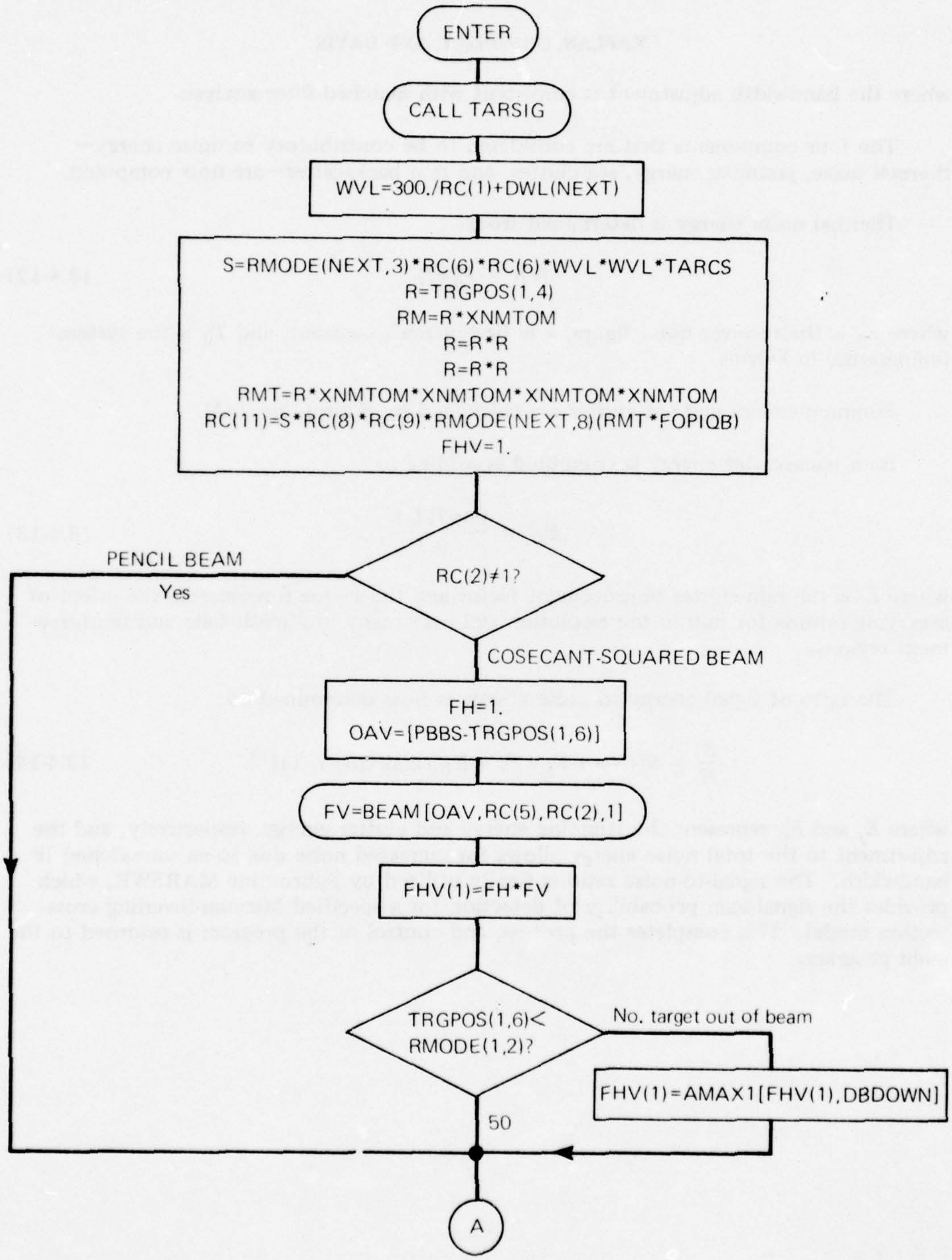


Fig. 3.4-1—DETPROB(PDSS) flowchart

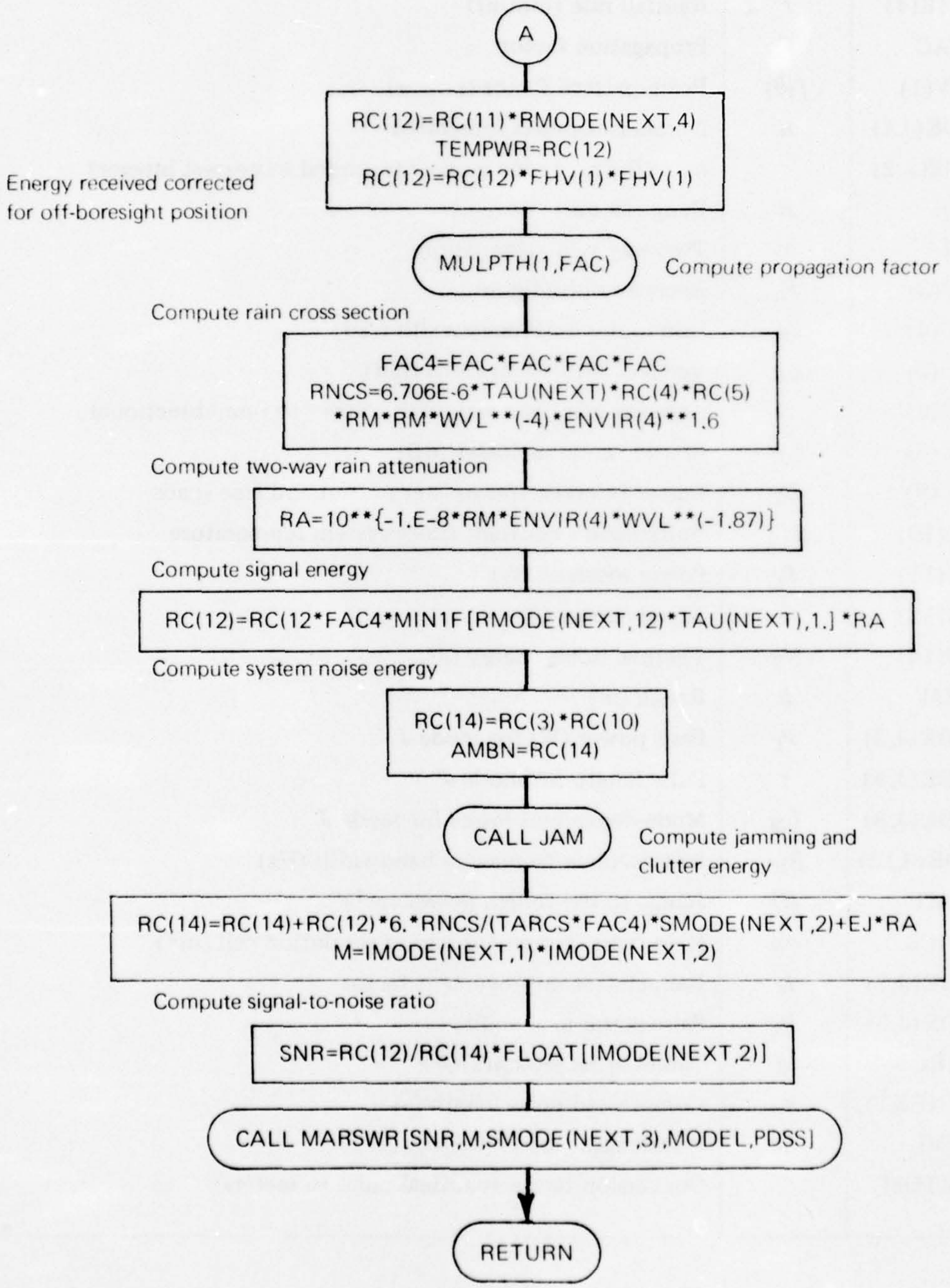


Fig. 3.4-1—DETPROB (PDSS) flowchart (Continued)

Table 3.4-2 — DETPROB Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
EJ	$E_J + E_c$	Jamming and sea-clutter energy
ENVIR(4)	r	Rainfall rate (mm/hr)
FAC	F	Propagation factor
FHV(1)	$f(\theta)$	Beam-pattern factor (power)
IMODE(J,1)	n	Number of pulses integrated
IMODE(J,2)		max ($B_{\text{IF}}\tau_c$, 1) for mode J (rounded to nearest integer)
R	R	Range (n.mi.)
RA	A_r	Two-way rain attenuation
RC(3)	F_n	Receiver noise figure
RC(4)	θ_B	Horizontal 3-dB beamwidth (rad)
RC(5)	ϕ_B	Vertical 3-dB beamwidth (rad)
RC(6)	G	Antenna gain, one-way with respect to omnidirectional
RC(8)	L_R	Receive antenna losses (dB)
RC(9)	L_T	Losses between transmitter output and free space
RC(10)	kT_0	Boltzmann's constant times system temperature
RC(11)	P_r	Power received (W)
RC(12)	S	Signal energy (J)
RC(14)	N_T	Thermal noise energy (J)
RM	R	Range (m)
RMODE(J,3)	P_t	Peak power (W) for mode J
RMODE(J,4)	τ	Pulse length for mode J
RMODE(J,8)	L_M	Mode-dependent losses for mode J
RMODE(J,12)	B_{IF}	Intermediate-frequency bandwidth (Hz)
RMT	R^4	Range to the fourth power (m^4)
RNCS	σ_R	Rain cross section for rain in resolution cell (m^2)
SMODE(J,2)	I_c	Rain-clutter improvement factor
SMODE(J,3)	P_f	False-alarm probability
TARCS	σ	Target cross section (m^2)
TAU(NEXT)	τ_c	Compressed-pulse length (s)
WVL	λ	Wavelength (m)
XNMTOM		Conversion factor (nautical miles to meters)

3.5 Subroutine ENVIRN

Subroutine ENVIRN is called by the SURSEM Executive Routine. It controls the input of the four environmental parameters: wind-speed, height of wind-speed measurement, multipath indicator, and rainfall rate. The subroutine also determines the standard deviation of the wave height by the method described below.

Burling [5] suggests the following relationships

$$\bar{H}_{1/3} = 4\sigma_h \tag{3.5-1}$$

where $\bar{H}_{1/3}$ is the significant wave height and σ_h is the standard deviation. The significant wave height is related to the wind speed at the sea surface by

$$\bar{H}_{1/3} = 2.667 \times 10^{-2}\, V^2 \tag{3.5-2}$$

where V and $\bar{H}_{1/3}$ are given in meters per second and meters, respectively.

Pierson [2] has shown that the windspeed at 10 meters above the surface is related to the speed at greater heights by

$$V_{10} = V_H \left(\frac{10}{H}\right)^{0.09682} \tag{3.5-3}$$

where V_H is the windspeed as measured at height H above the sea. Combining Eqs. (3.5-1), (3.5-2) and (3.5-3) yields (substituting V_{10} for V)

$$\sigma_h = 0.00667\, (V_H)^2 \left(\frac{H}{10}\right)^{-0.19364} . \tag{3.5-4}$$

Before returning control of the program to the SURSEM Executive Routine, ENVIRN also clears the initial and final altitudes of the clutter cell to zero, sets the initial game time to the time the target leaves its initial position, and sets an indicator that designates the target as being originally below the horizon.

The dictionary of variables used in Subroutine ENVIRN is given in Table 3.5-1. The ENVIRN flowchart is shown in Fig. 3.5-2.

Table 3.5-1 — ENVIRN Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ENDTIME		Time limit that ends run (s)
ENVIR(1)	V_H	Wind speed at height H (knots)
ENVIR(2)	H	Height of wind-speed measurement (10^3 ft)
SIGMAH	σ_h	Standard deviation of the wave height (m)
T	t	Game time (hr)
TRGPOS(1,4)	R	Target range, set to −1 in ENVIRN to indicate that the target is beyond the horizon
XYZF(NUMTGT,3)		Final altitude of clutter patch (set to zero)
XYZI(NUMTGT,3)		Initial altitude of clutter patch (set to zero)

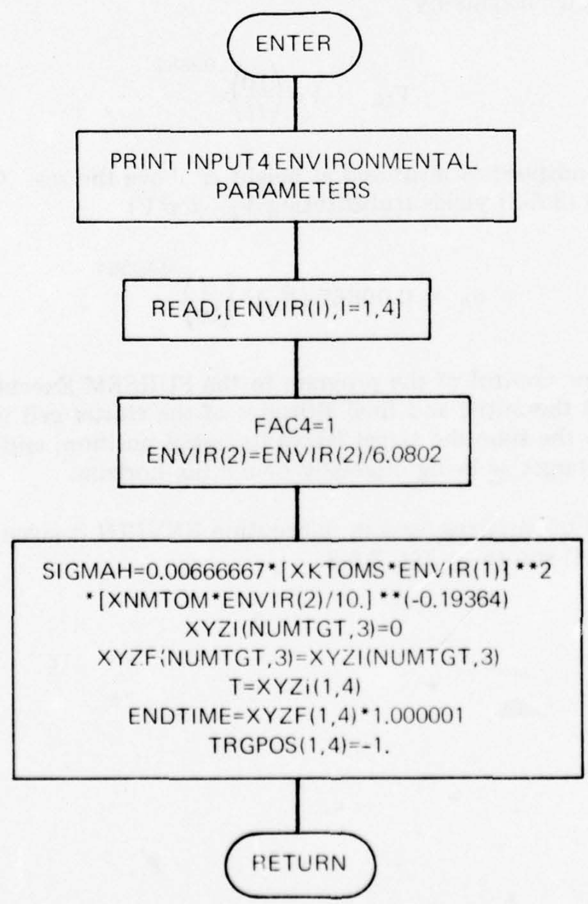


Fig. 3.5-2—ENVIRN flowchart

3.6 Subroutine GAIN

Subroutine GAIN is called by Subroutines MULPTH and JAM. Its primary function is to determine the field strength ratio in the direction of direct and reflected rays.

The program is called with the indicator IKEY specifying which ray is under consideration (see Table 3.6-1 and Fig. 3.6-2 for the dictionary of variables and flowchart, respectively). If a direct ray is being considered, the field strength ratio is determined by taking the square root of the beam pattern function that is calculated in either JAM or MULPTH. Control of the program is then returned to the calling subroutine.

For an indirect ray, the difference in azimuth between the first target and the target called for (ITAR) is determined and used by the BEAM function to calculate the horizontal beam pattern factor (f_{PH}). The angle between the direct ray and the reflected ray is then calculated according to (see Fig. 3.12-3)

$$\sin \alpha_v = \frac{2R_2 \sin \psi \cos \psi}{R} .$$

Table 3.6-1 — GAIN Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ALFV	α_v	Angle between direct and reflected rays at antenna (rad)
FH	f_{PH}	Horizontal beam pattern factor (power)
FHV(K)		Pattern function (total) for target K
FV	f_{PV}	Vertical beam pattern factor (power)
GAINR	$f(\beta)$	Ratio of field strength in direction of specified ray to the field strength in the beam-maximum direction
IKEY		Indicator in calling sequence (0 = direct ray, 1 = reflected ray)
ITAR		Target number
OAH		Horizontal angle between boresight and target (rad)
OAV		Vertical angle between boresight and target (rad)
RC(2)		Beam pattern indicator (0 = pencil beam, 1 = $\csc^2$ beam)
RC(4)		Horizontal 3-dB beamwidth (rad)
RC(5)		Vertical 3-dB beamwidth (rad)
RC(7)		One-way sidelobe level (v)
SRTAR	R_2	Slant range from target to reflection point (m)
TRGPOS(I,4)	R	Slant range of target I (m)
TRGPOS(I,5)		Target I azimuth (rad)
TRGPOS(I,6)	θ	Target I elevation with respect to the horizon (rad)

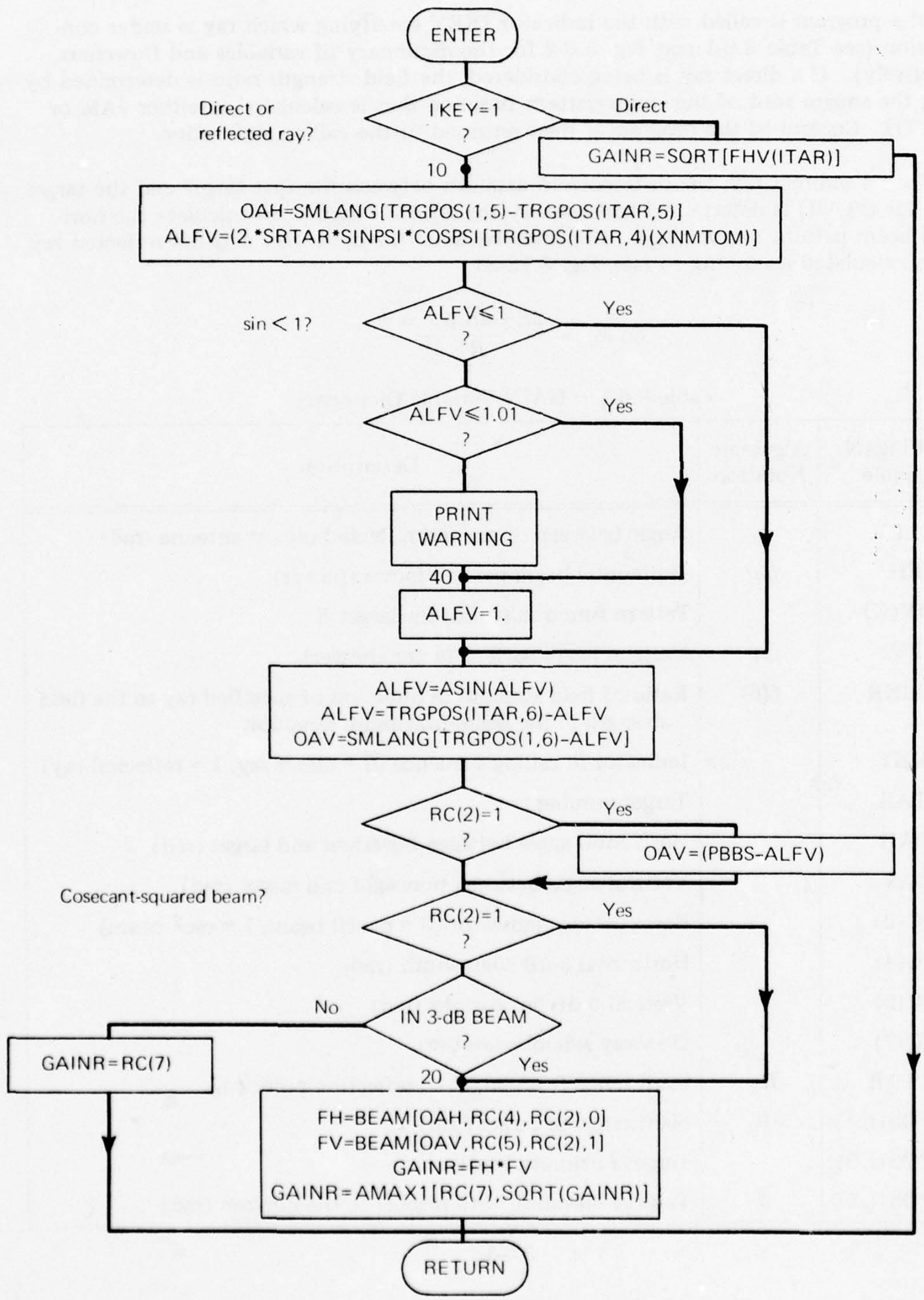


Fig. 3.6-2—GAIN(IKEY,ITAR,GAINR) flowchart

This information is used to calculate the vertical angle between boresight and target. For the cosecant-squared beam, this amounts to the angular difference between pencil-beam boresight and the target. For the pencil beam, the beam is assumed to be on the first target at all times, and the vertical angle between the first target and the target under consideration is determined. This information is used by function BEAM to calculate the vertical beam pattern factor f_{PV}, which is used in turn to calculate the total beam pattern factor from

$$f_P = f_{PH} f_{PV}$$

and the field strength ratio from

$$f(\beta) = f_P^{1/2} .$$

If this value exceeds the first sidelobe level, it is retained; otherwise, the field strength ratio is assigned the value of the first sidelobe level. The first sidelobe level is also assigned to the field strength ratio if the target is not within the 3-dB beamwidth in the pencil-beam case.

When the calculation of the field strength ratio has been completed, control of the program is returned to the calling subroutine.

3.7 Function GAM

Function GAM is called by several statements in Subroutine MARSWR. It is used to evaluate the incomplete gamma function of the form

$$f(a,p) = \int_0^a \frac{e^{-u}u^p}{p!}\,du \tag{3.7-1}$$

$$f(a,p) = 1 - \int_a^\infty \frac{e^{-u}u^p}{p}\,du\,. \tag{3.7-2}$$

The evaluation is completed as follows: integration by parts yields

$$f(a,p) = \frac{e^{-a}a^p}{p!} + \int_0^a \frac{e^{-u}u^{p-1}}{(p-1)!}\,du\,. \tag{3.7-3}$$

This is a recursive process and leads to

$$f(a,p) = 1 - \sum_{k=0}^{\infty} \frac{e^{-a}a^k}{k!}\,, \tag{3.7-4}$$

or equivalently to

$$f(a,p) = \sum_{k=p+1}^{\infty} \frac{e^{-a}a^k}{k!}\,. \tag{3.7-5}$$

Equations (3.7-4) and (3.7-5) are the expressions that are then used by GAM to evaluate the incomplete gamma function. The choice of Eq. (3.7-4) or (3.7-5) is determined by the convergence characteristics of the respective summations.

It has been found that Eq. (3.7-4) converges faster than does Eq. (3.7-5) when $a \geqslant p$, whereas the reverse is true when $a < p$. Selection is made on the basis of this criterion, and an iterative loop is set up that calculates each term of the summation. As the program is currently configured, the loop is terminated whenever the value of an single term in the summation is less than the accuracy of the computer that is exercising the program. The loop involving Eq. (3.7-4) is also terminated when $k = p$.

After the iterative loop is terminated, the value of GAM is determined and control of the program is returned to MARSWR.

A dictionary of the variables used in function GAM is given in Table 3.7-1. The flowchart is shown in Fig. 3.7-2.

Table 3.7-1 — GAM Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
B	a	Parameter used in the incomplete gamma function
GAM	$f(a, p)$	Value of incomplete gamma function corresponding to a and p
N	p	Parameter used in the incomplete gamma function
TEMP		Partial sum of $\Sigma_{k=0}^{p} e^{-a}a^k/k!$ or $\Sigma_{k=p+1}^{\infty} e^{-a}a^k/k!$
TERM		The kth term in the above summations

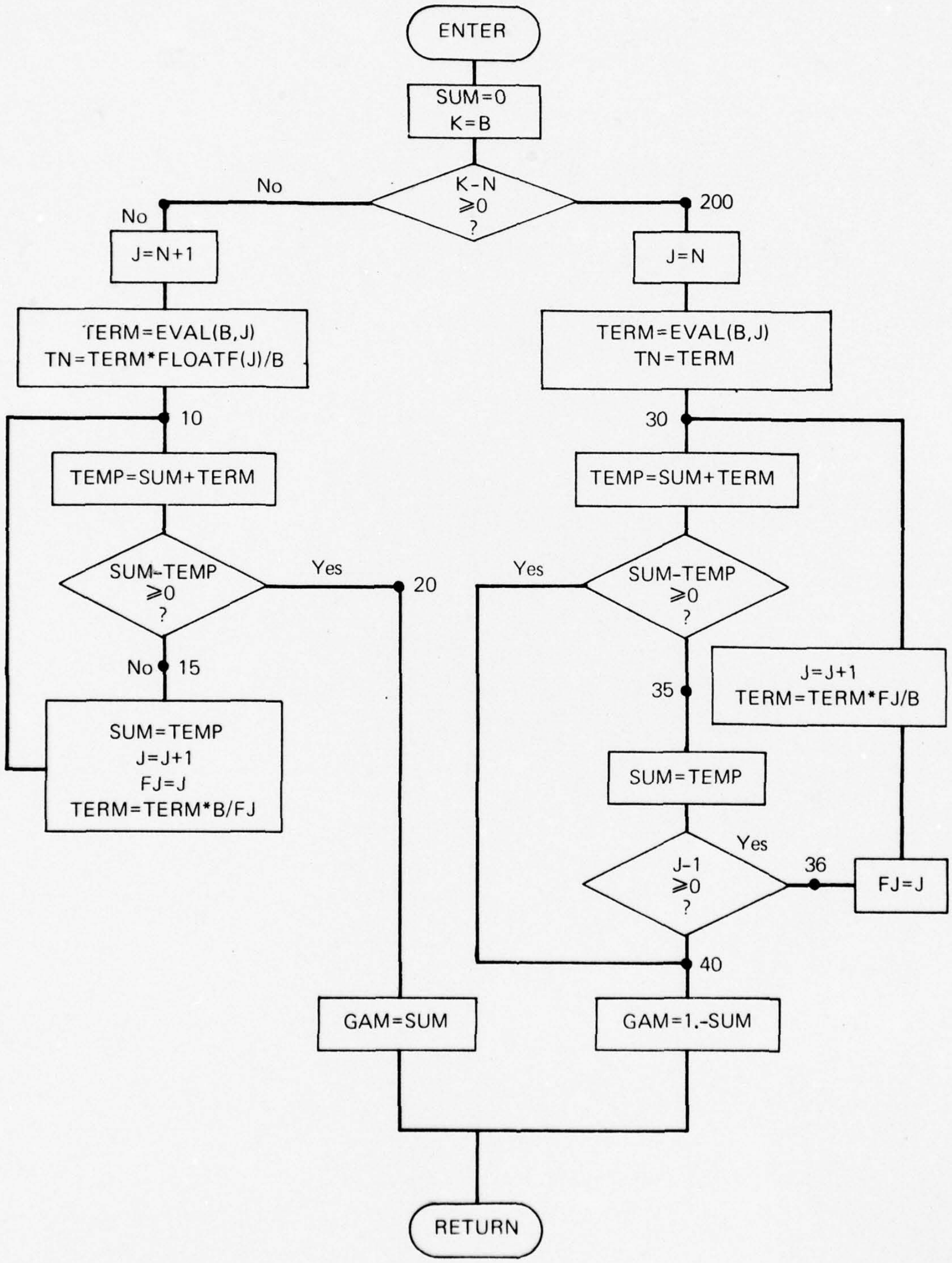


Fig. 3.7-2—GAM(B,N,TN) flowchart

3.8 Subroutine INITIAL

Subroutine INITIAL is called by the SURSEM Executive Routine. Its primary purpose is to establish constants, read in data, and convert variables to the correct units for utilization by the other subroutines. (See Table 3.8-1 and Fig. 3.8-2 for the dictionary of variables and flowchart.)

Table 3.8-1 — INITIAL Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
DWL(J)		Frequency increment (MHz), mode J
IMODE(J,1)		Number of pulses integrated
IMODE(J,2)		max $(B_{IF}\tau_c, 1)$ rounded to the nearest integer
MILLION		1×10^6
MM		Effective number of pulses integrated
NSCAN		Number of scan modes
PI	π	3.1415926536
PIOVER2	$\pi/2$	
POLRZ		Linear polarization (deg)—(0° = horizontal, 90° = vertical)
RADIAN		57.29578 deg
RC(1)	f	Radar frequency (MHz)
RC(2)		Antenna pattern function indicator (0 = pencil beam, 1 = $\csc^2$ beam)
RC(3)		Receiver noise
RC(4)		Horizontal 3-dB beamwidth (degrees to radians)
RC(5)		Vertical 3-dB beamwidth (degrees to radians)
RC(6)		One-way antenna gain
RC(7)		One-way sidelobe level
RC(8)		Receiver losses
RC(9)		Transmitter losses
RC(10)	kT	Boltzmann's constant times system temperature (W-s)
RMODE(J,1)		Lower limit of elevation angle coverage (deg) for mode J
RMODE(J,2)		Upper limit of elevation angle coverage (deg) for mode J
RMODE(J,3)		Peak power (megawatts to watts)
RMODE(J,4)		Pulse length (microseconds to seconds)
RMODE(J,5)		Interlook period (seconds to hours)
RMODE(J,6)		Scan offset (seconds to hours)
RMODE(J,7)		Instrumented range (n.mi.)
RMODE(J,8)		Mode-dependent losses
RMODE(J,11)		Compressed-pulse length (μs) for mode J
RMODE(J,12)		Intermediate-frequency bandwidth (megahertz to hertz) for mode J
SM		Blanking time (μs)
SMODE(J,1)		Blanking range (n.mi.)
SMODE(J,2)		Rain-clutter improvement factor
SMODE(J,3)		Equivalent to XMPF
SUBC(J)		Sea-clutter improvement factor
TAU	τ_c	Compressed-pulse length (s) for mode J
TWOPI	2π	
XMPF		Negative $\log_{10}$ of the false-alarm probability

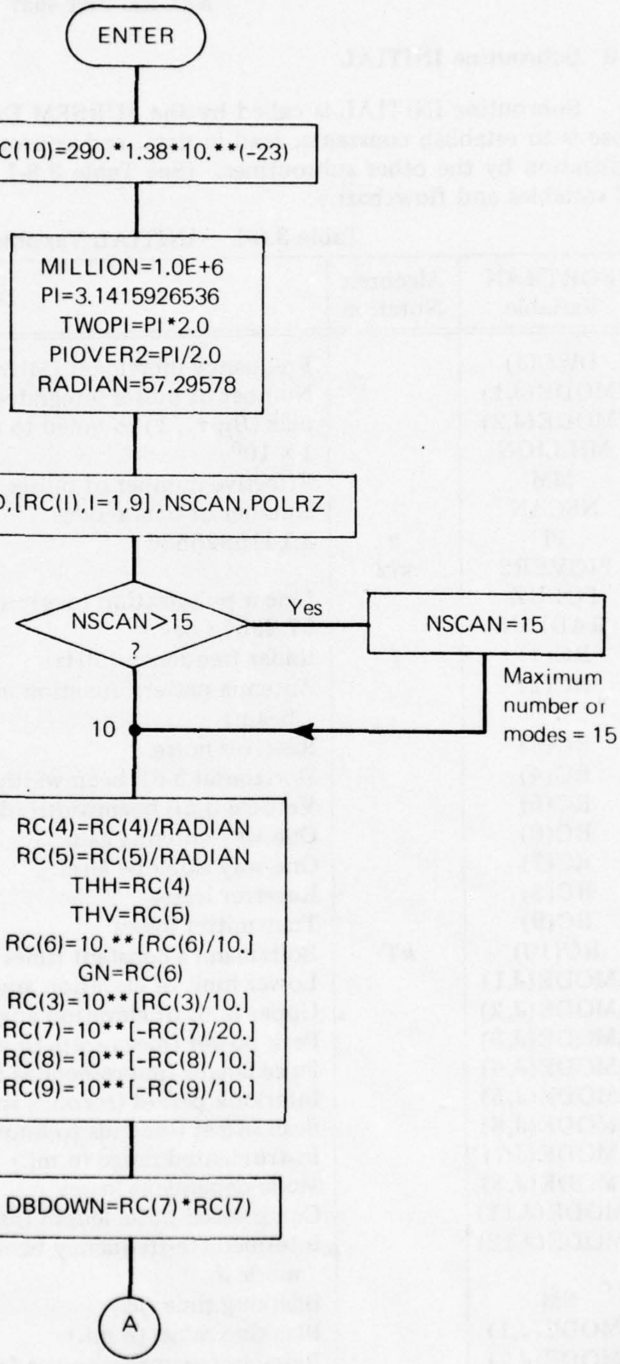


Fig. 3.8-2—INITIAL flowchart

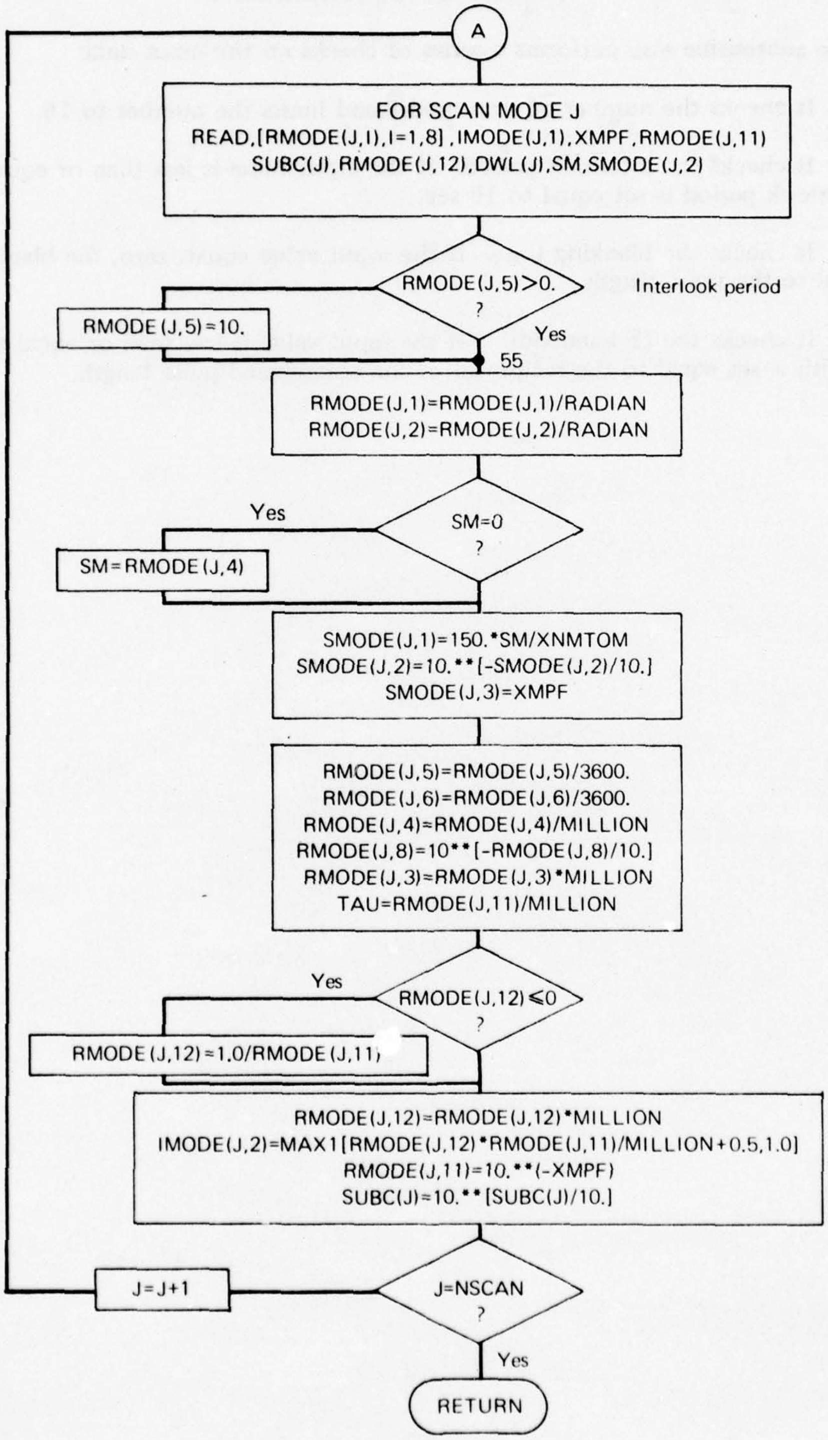


Fig. 3.8-2—INITIAL flowchart (Continued)

The subroutine also performs a series of checks on the input data:

1. It checks the number of scan modes and limits the number to 15.

2. It checks the interlook period. If the input value is less than or equal to zero, the interlook period is set equal to 10 sec.

3. It checks the blanking time. If the input value equals zero, the blanking time is set equal to the pulse length.

4. It checks the IF bandwidth. If the input value is less than or equal to zero, the bandwidth is set equal to the reciprocal of the compressed pulse length.

3.9 Subroutine JAM

Subroutine JAM is called by Subroutine DETPROB to determine the magnitude of received jamming and sea-clutter energy. The subroutine can account for both self-screening and standoff jammers.

After initializing the required variables, the subroutine considers the jamming energy transmitted from each target. For each jammer a beam pattern factor must be determined to account for the jammer's being off beam center. If the antenna beam pattern has been designated as a pencil beam, the antenna beam pattern (power) is approximated by a $|(\sin kx)/kx|^2$ curve in both the horizontal and vertical directions. In this case a check is made to determine whether the jammer is beyond the first null. Jammers that are not inside the first null are assigned a corresponding beam pattern factor that is equal to the sidelobe level. For those jammers whose angular position places them within the first null, the function BEAM is used to determine a horizontal (f_H) and vertical (f_V) beam pattern factor. The total beam pattern factor is then given by

$$f_{HV} = f_H / f_V . \qquad (3.9\text{-}1)$$

A similar procedure is followed with the cosecant-squared beam.

The free-space jamming energy for each jammer is now found from

$$E_{OJ} = \frac{G_r L_r L_m S_j f_{HV} \lambda^2}{(4\pi^2) R^2} \qquad (3.9\text{-}2)$$

where G_r is the one-way antenna gain, L_r is the receiving antenna loss, L_m is the mode-dependent loss, S_j is the jamming power density, λ is the radar wavelength, and R is the slant range from radar to target.

Subroutine MULPTH is called to account for multipath effects. It calculates a one-way amplitude propagation factor F, which is used to determine individual jamming energy as follows:

$$E_J = E_{OJ} F^2 . \qquad (3.9\text{-}3)$$

Total jamming energy is given by

$$E_{TJ} = \sum_{1}^{N_J} E_J \qquad (3.9\text{-}4)$$

where N_J is the number of jammers.

The total sea-clutter energy is determined when all jammers have been considered. The first step in the determination of sea-clutter energy is the computation of the normalized mean backscatter σ_0. This is performed by Subroutine CLUTSIG, which evaluates σ_0 as a function of radio frequency, Beaufort sea state, incidence angle, and polarization orientation. The Beaufort sea state is calculated from the input wind velocity and

the height at which the velocity measurement is assumed to take place [2]. An equivalent wind speed at a height of 10 m is found from

$$V_{10} = V_H \left(\frac{7.5}{H}\right)^{0.09682} . \tag{3.9-5}$$

The Beaufort sea state is then given by

$$B_{SS} = 0.6077(V_{10})^{0.7186} . \tag{3.9-6}$$

The other parameters required by CLUTSIG (i.e., radio frequency and incidence angle) are inputs to Subroutine JAM.

When the normalized mean backscatter has been evaluated, a differential clutter element of area $r d\phi\, dr$ is considered (see Fig. 3.9-1). The energy received (dE) from this element is given by

$$dE = C \frac{G^2}{r^4} r d\phi\, dr \tag{3.9-7}$$

where

$$C = \frac{P_T \lambda^2 L_r L_t L_m \sigma_0}{4\pi^3} \tag{3.9-8}$$

and G, the antenna directional gain, is a function of the horizontal and vertical angular displacement of the clutter cell with respect to the beam center; that is,

$$G = G(\theta_T + \alpha, \phi) . \tag{3.9-9}$$

Here θ_T is the angle between the local horizontal and the target or beam center (see Fig. 3.9-2)

$$\theta_T \approx \sin^{-1}\left(\frac{h_T - h_r}{R} - \frac{R}{2a}\right), \tag{3.9-10}$$

and α is the angle between the local horizontal and the reflected ray; to a first-order approximation α is given by

$$\alpha \approx \sin^{-1}\left(\frac{h_r}{R} - \frac{R}{2a}\right); \tag{3.9-11}$$

ϕ is the azimuth of the clutter element with respect to beam center.

The total sea-clutter energy is now considered to be the energy that is reflected from an annulus (see Fig. 3.9-3) of width ΔR given by

$$\Delta R = 1/2 c\tau \sec \psi \tag{3.9-12}$$

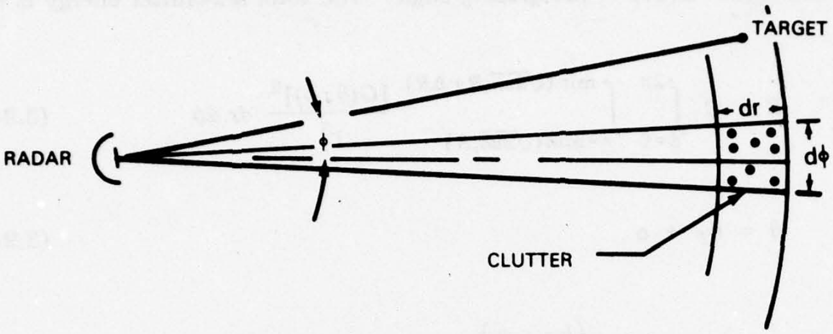


Fig. 3.9-1—Differential clutter element

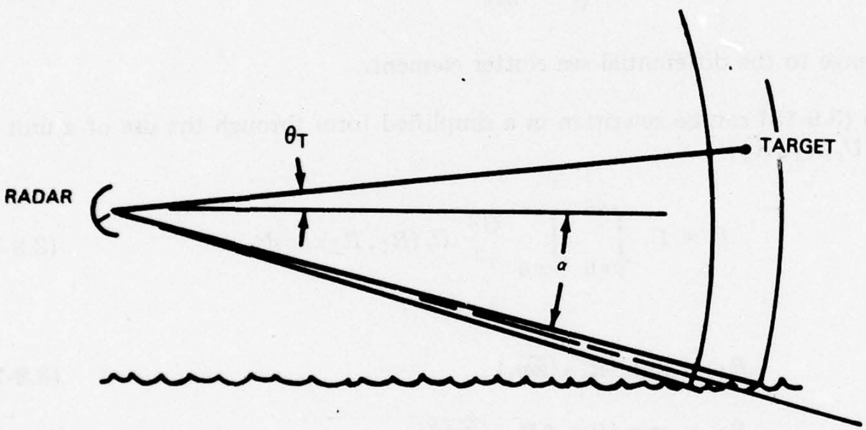


Fig. 3.9-2—Geometry of clutter element

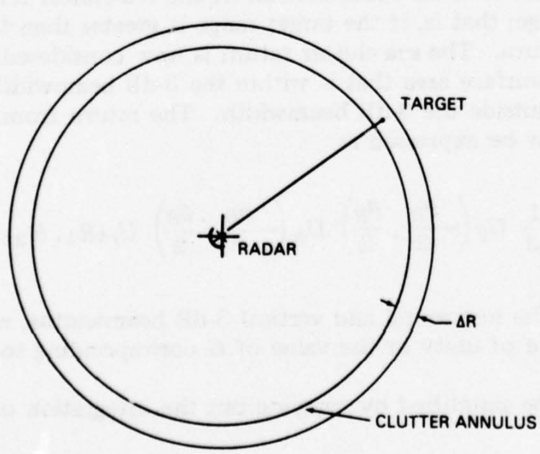


Fig. 3.9-3—Annulus of total sea clutter

where τ is the pulse width and ψ is the grazing angle. The total sea-clutter energy is expressed as

$$E = C \int_{\phi=0}^{2\pi} \int_{r=\min(\sqrt{2ah},R)}^{\min(\sqrt{2ah},R+\Delta R)} \frac{[G(\theta,\phi)]^2}{r^3}\, dr\, d\phi \qquad (3.9\text{-}13)$$

where

$$\theta = \theta_T + \alpha , \qquad (3.9\text{-}14)$$

or

$$\theta = \theta_T + \sin^{-1}\left(\frac{h}{r} - \frac{r}{2a}\right) \qquad (3.9\text{-}15)$$

and r is the range to the differential sea-clutter element.

Equation (3.9-13) can be rewritten in a simplified form through the use of a unit gate function $U_r(R_1, R_2)$:

$$E = C \int_{\phi=0}^{2\pi} \int_{r=0}^{\infty} \frac{G^2}{r^3}\, U_r(R_1, R_2)\, dr\, d\phi \qquad (3.9\text{-}16)$$

where

$$R_1 = \min(R, \sqrt{2ah}) , \qquad (3.9\text{-}17)$$

$$R_2 = \min(R + \Delta R, \sqrt{2ah}) . \qquad (3.9\text{-}18)$$

Essentially this procedure restricts consideration of the sea-clutter return to that from within the horizon range; that is, if the target range is greater than the horizon range, there is no clutter return. The sea-clutter return is now considered as coming from two distinct areas: (a) The surface area that is within the 3-dB beamwidth and (b) the area of the annulus that is outside the 3-dB beamwidth. The return from the area within the 3-dB beamwidth can now be expressed as

$$E_1 = C \int_{\phi=0}^{2\pi} \int_{r=0}^{\infty} \frac{1}{r^3}\, U_\theta\left(-\frac{\theta_B}{2}, \frac{\theta_B}{2}\right) U_\phi\left(-\frac{\phi_B}{2}, \frac{\phi_B}{2}\right) U_r(R_1, R_2)\, dr\, d\phi \qquad (3.9\text{-}19)$$

where ϕ_B and θ_B represent the horizontal and vertical 3-dB beamwidths, respectively, and G has been assigned the value of unity or the value of G corresponding to beam center.

Equation (3.9-19) can be simplified by carrying out the integration on ϕ:

$$E_1 = C\phi_B \int_{r=0}^{\infty} \frac{1}{r^3}\, U_\theta\left(-\frac{\theta_B}{2}, \frac{\theta_B}{2}\right) U_r(R_1, R_2)\, dr . \qquad (3.9\text{-}20)$$

From Eq. (3.9-15) it is possible to express r as a function of θ:

$$r(\theta) = \frac{2h}{\sin(\theta - \theta_T) + [\sin^2(\theta - \theta_T) - (2h/a)]^{1/2}}. \tag{3.9-21}$$

Thus the integral of Eq. (3.9-20) can be expressed entirely as a function of r:

$$E_1 = C\phi_B \int_{r=0}^{\infty} \frac{1}{r^3} U_r \left[r\left(-\frac{\theta_B}{2}\right), \quad r\left(\frac{\theta_B}{2}\right)\right] U_r(R_1, R_2)\, dr\,, \tag{3.9-22}$$

or

$$E_1 = C\phi_B \int_{r=0}^{\infty} \frac{1}{r^3} U_r \left\{ \max \left[R_1, r\left(-\frac{\theta_B}{2}\right)\right], \quad \min \left[R_2, r\left(\frac{\theta_B}{2}\right)\right]\right\} dr\,. \tag{3.9-23}$$

This can be simplified with the following definitions:

$$S_1 \equiv \max \left[R_1, r\left(-\frac{\theta_B}{2}\right)\right], \tag{3.9-24}$$

$$S_2 \equiv \min \left[R_2, r\left(\frac{\theta_B}{2}\right)\right]. \tag{3.9-25}$$

Equation (3.9-23) can now be written as

$$E_1 = C\phi_B \int_{r=0}^{\infty} \frac{1}{r^3} U_r(S_1, S_2)\, dr\,. \tag{3.9-26}$$

Integration of Eq. (3.9-26) yields

$$E_1 = \frac{1}{2} C\phi_B \left(\frac{1}{S_1^2} - \frac{1}{S_2^2}\right). \tag{3.9-27}$$

Similarly the energy from the remainder of the annulus can be expressed as

$$E_2 = CS\left[\pi\left(\frac{1}{R_1^2} - \frac{1}{R_2^2}\right) - H\right] \tag{3.9-28}$$

where S represents a constant value assigned to the antenna directional gain (usually the sidelobe level) and

$$H = \frac{1}{2} \phi_B \left(\frac{1}{S_1^2} - \frac{1}{S_2^2}\right). \tag{3.9-29}$$

The total sea-clutter energy can now be expressed according to

$$E = E_1 + E_2 = C\left[5\pi\left(\frac{1}{R_1^2} - \frac{1}{R_2^2}\right) - H(S - 1)\right]. \tag{3.9-30}$$

The sea-clutter energy is further reduced by the input sea-clutter improvement factor I_c,

$$E_C = EI_C, \tag{3.9-31}$$

and the total jamming energy plus clutter energy is found from

$$E_{J+C} = E_{TJ} + E_C . \tag{3.9-32}$$

The variables used in Subroutine JAM are listed in Table 3.9-3. The flowchart is shown in Fig. 3.9-4.

Table 3.9-3 — JAM Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ALPHA	α	Grazing angle of clutter cell (rad)
ALPHAD	α	Grazing angle of clutter cell (deg)
ACON		Constant used in sea-state calculation
BEAUS		Beaufort sea state
BETA		Constant used in sea-state calculation
DBDOWN		One-way sidelobe level
DC		Distance to clutter cell (on 4/3 Earth's radius) (m)
DSTAR		Distance to radar horizon (n.mi.)
DWL(J)		Incremental change to RF for mode J
EJ	E_J	Jamming and sea-clutter energy (J)
ENVIR(1)		Wind speed (knots)
ENVIR(2)	V_H	Height of wind-speed measurement (10^3 ft)
FH	f_H	Normalized horizontal beam pattern factor (power)
FHV(J)	f_{HV}	Total normalized beam pattern factor for jammer J (power)
FOPISQ	$4\pi^2$	Constant
FV	f_V	Normalized vertical beam pattern factor (power)
HR	h_r	Height of radar (m)
HT	h_T	Height of target under consideration (m)
NUMTGT		Number of targets to be considered plus one
PAH		Horizontal angle between target being tracked and jamming target (rad)

Table 3.9-3 – JAM Variable Dictionary (Continued)

FORTRAN Variable	Algebraic Notation	Description
PAV		Vertical angle between target being tracked and jamming target (rad)
PBBS		Pencil-beam boresight elevation with respect to the horizon (rad)
PHIB	ϕ_B	Horizontal 3-dB beamwidth (rad)
PJ	S_J	Jamming power density (W/Hz)
POLRZ		Linear polarization (deg)
R1	R_1	min (slant range, horizontal range)
R2	R_2	max (slant range plus pulse width, horizontal range)
RC(1)		Radar frequency (Hz) for mode 1
RC(2)		Antenna pattern function indicator (0 = pencil beam, 1 = $\csc^2$ beam)
RC(4)	ϕ_B	Horizontal 3-dB beamwidth (rad)
RC(5)	θ_B	Vertical 3-dB beamwidth (rad)
RC(6)	G_r	One-way antenna gain (dB)
RC(8)	L_r	Receiver antenna loss (dB)
RC(9)	L_t	Losses between transmitter output and free space (dB)
RPTB	$r(\theta_B/2)$	Range corresponding to $\theta_B/2$ (m)
RE	a	4/3 of the Earth's radius (m)
RMODE(J,2)		Upper boundary of elevation-angle coverage (rad)
RMODE(J,8)	L_m	Mode-dependent losses for scan mode (J)
RMTB	$r(-\theta_B/2)$	Range corresponding to $-\theta_B/2$
SIGC	E_c	Total sea-clutter energy adjusted by improvement factor (J)
SIGJAM	S_J	Jamming power density (W/Hz)
SIGZ	σ_0	Normalized mean backscatter (m^2/m^2)
SINALF	$\sin \alpha$	Sine of grazing angle at clutter cell
SR	R	Slant range of target (m)
SUBC(J)	I_c	Clutter improvement factor for mode J
TEMPWR		Signal energy with target at center of beam
THETB	θ_B	Vertical 3-dB beamwidth (rad)
THETT	θ_T	Target elevation with respect to the local horizontal (rad)
TRGPOS	R	Slant range of target (n.mi.)
TRGPOS(J,3)		Height of target J (n.mi.)
TRGPOS(J,4)		Slant range of target J (n.mi.)
TRGPOS(J,5)		Azimuth angle of target J (rad)
TRGPOS(J,6)		Elevation angle of target J (rad)
V	$1/2c\tau \sec \psi$	Range extent of clutter cell (m)
WVL	λ	Wavelength (m)
WS	H	Intermediate parameter
XFRE		RF of scan mode under consideration (MHz)
XJAMFA	E_J	Jamming energy from jth jammer (J)

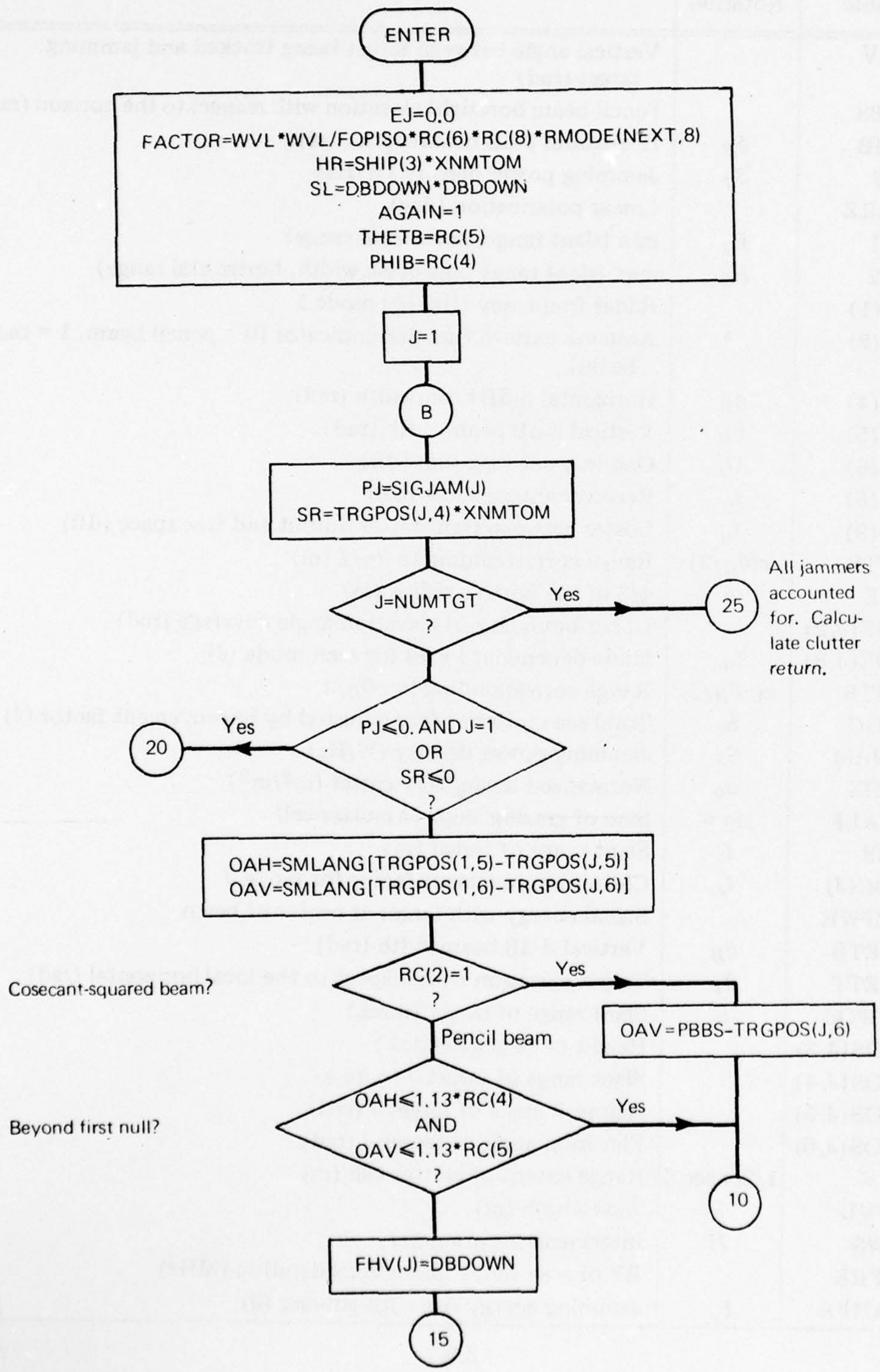


Fig. 3.9-4—JAM(EJ) flowchart

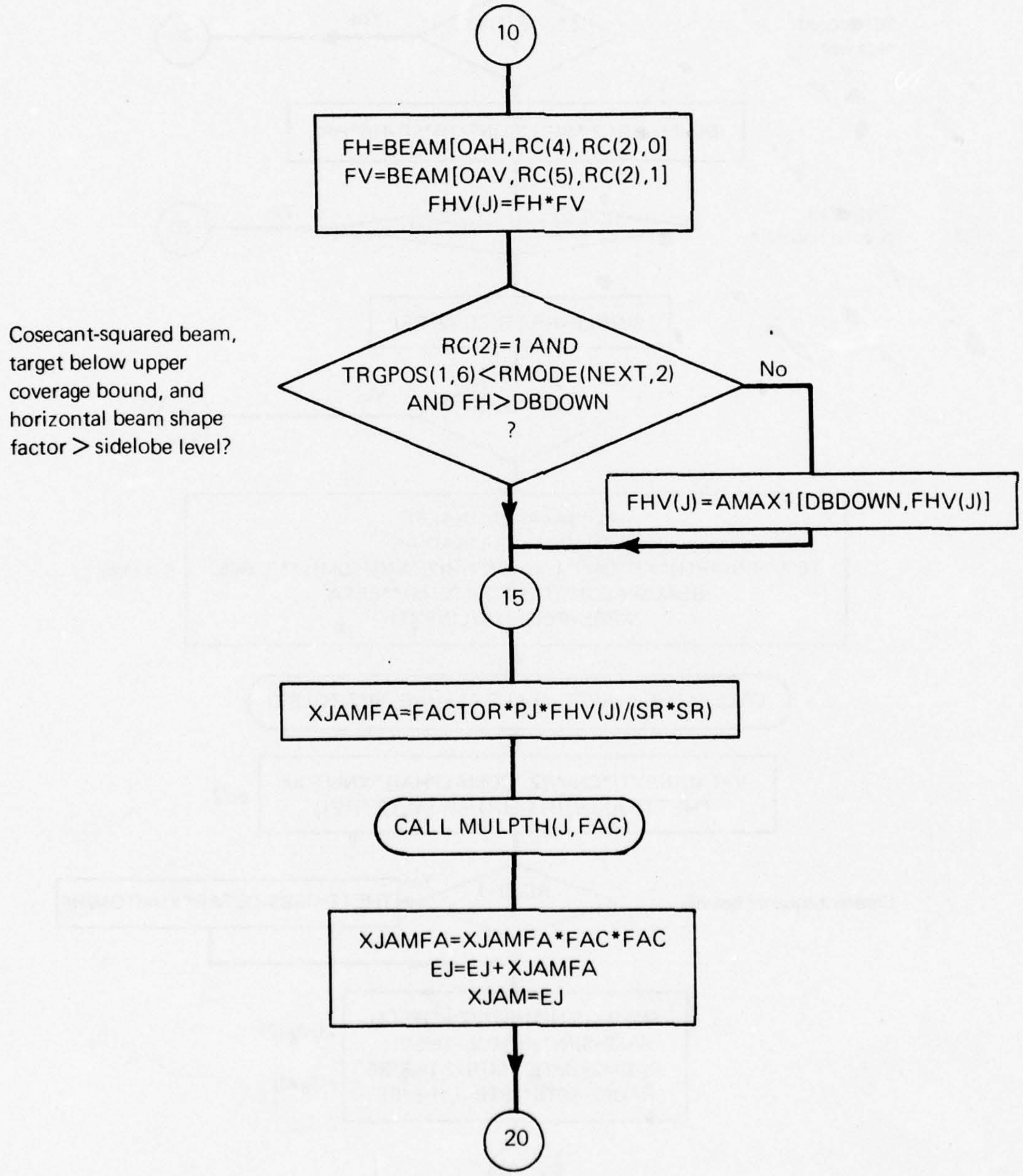


Fig. 3.9-4—JAM(EJ) flowchart (Continued)

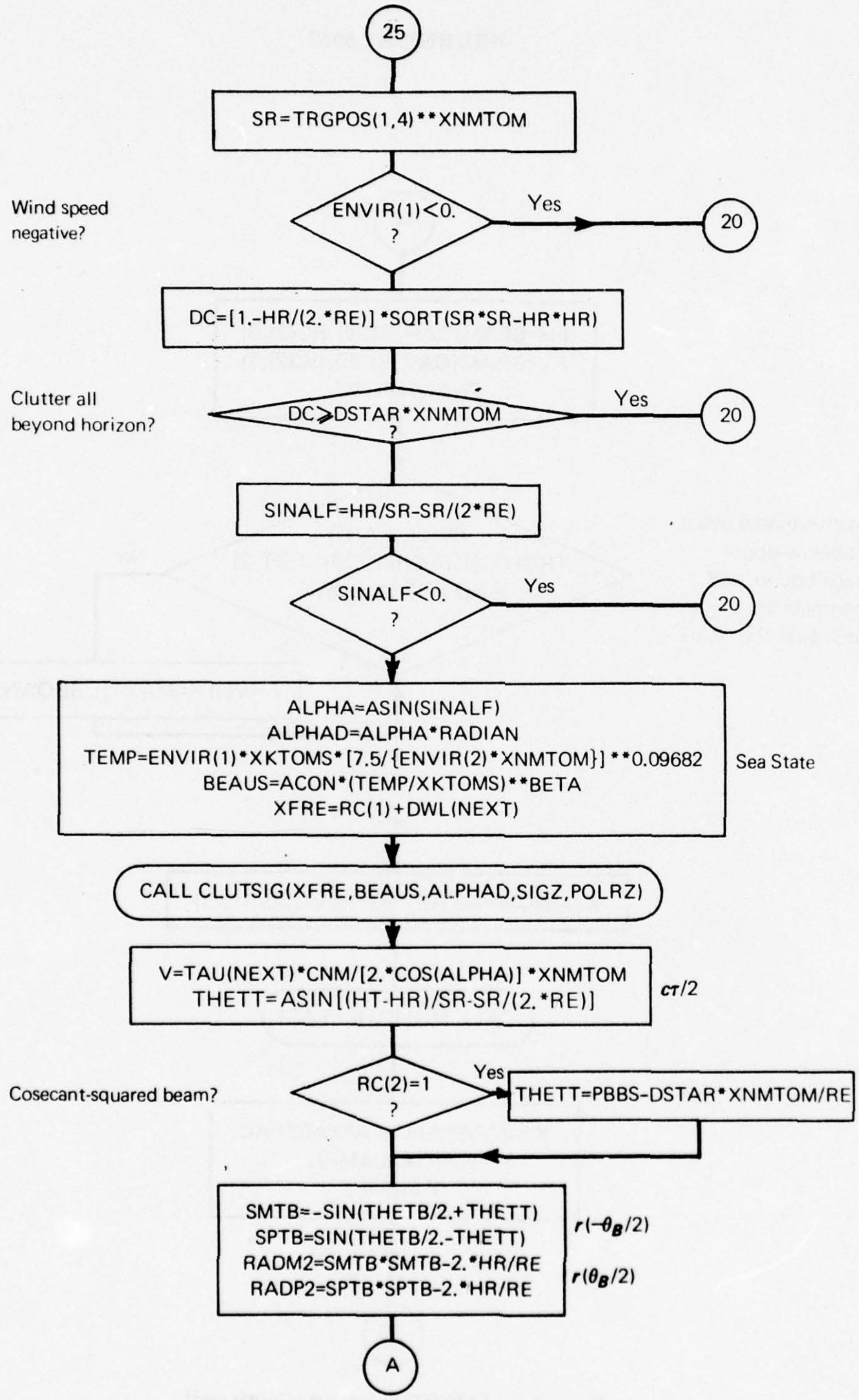


Fig. 3.9-4—JAM(EJ) flowchart (Continued)

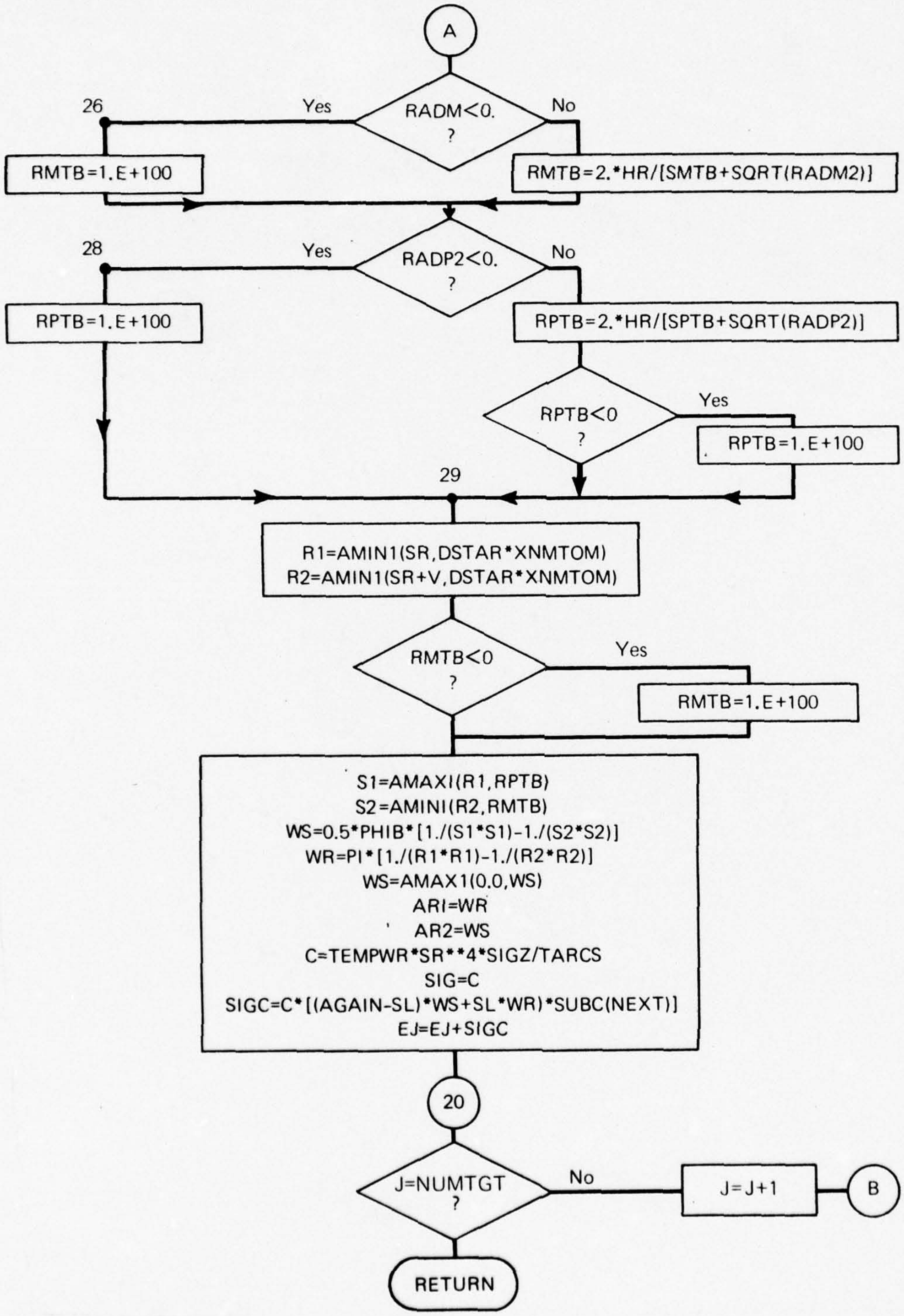


Fig. 3.9-4—JAM(EJ) flowchart (Continued)

3.10 Subroutine MARSWR

Subroutine MARSWR is called by Subroutine DETPROB. Its function is to calculate the probability of detection, given the signal-to-noise ratio, the number of pulses integrated, the probability of false alarm, and the Marcum-Swerling target model. The dictionary of variables and flowchart are given in Table 3.10-1 and Fig. 3.10-2.

Table 3.10-1 — MARSWR Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
DEVAL(A,B,C)		A double-precision function that evaluates e^{-Ay^B}/e^C
DGAM(X, Y)		A double-precision function that determines a value of the incomplete gamma function with parameters X and Y
EO	$f'(Y_j)$	Current value of derivative of $f(Y)$
E1	$f'(Y_{j+1})$	Incremented value of derivative
EN	N	Number of pulses integrated
ENPR	P_{FA}	Probability of false alarm
EVAL(A,B)		Single-precision version of DEVAL
FA		False-alarm probability expressed as an absolute value of a power of 10
FAN	$\log_{10} n'$	Logarithm of the false-alarm number
FAPREV		Probability of false alarm on previous pass through subroutine
FLOATF(J)		A function which changes an integer J to floating-point form
GAM(X,Y)		Single-precision version of DGAM(X,Y)
GAMPR	$f(Y_j)$	Current value of $f(Y)$
GS		$\Sigma_{j=0}^{N-1+k_{\max}} e^{-Y_b} Y_b^j/j!$
H	ΔY	Increment on the bias level for use with Huen's method
KASE		Swerling fluctuation model identifier (KASE = 0 for the non-fluctuating case)
KS	$k_{\max}$	Maximum value of summation parameter
N	N	Number of pulses integrated
NPREV		Number of pulses integrated on preceding pass through subroutine
PN	P_N	Probability of detection
PYB	$P_0^{1/n'}$	P_0 is the probability that there will not be a false alarm in the false-alarm time
SNR		Ratio of signal energy to noise energy
STEP	$f(Y_{j+1})$	Incremented value of $f(Y)$
SUMLOG(K)		A function that evaluates $\Sigma_{N=1}^{K} \log N$ for any integer K
X		Ratio of signal energy to noise energy
YB	Y_b	Bias level
YBPR	Y_0	Estimated value of bias level

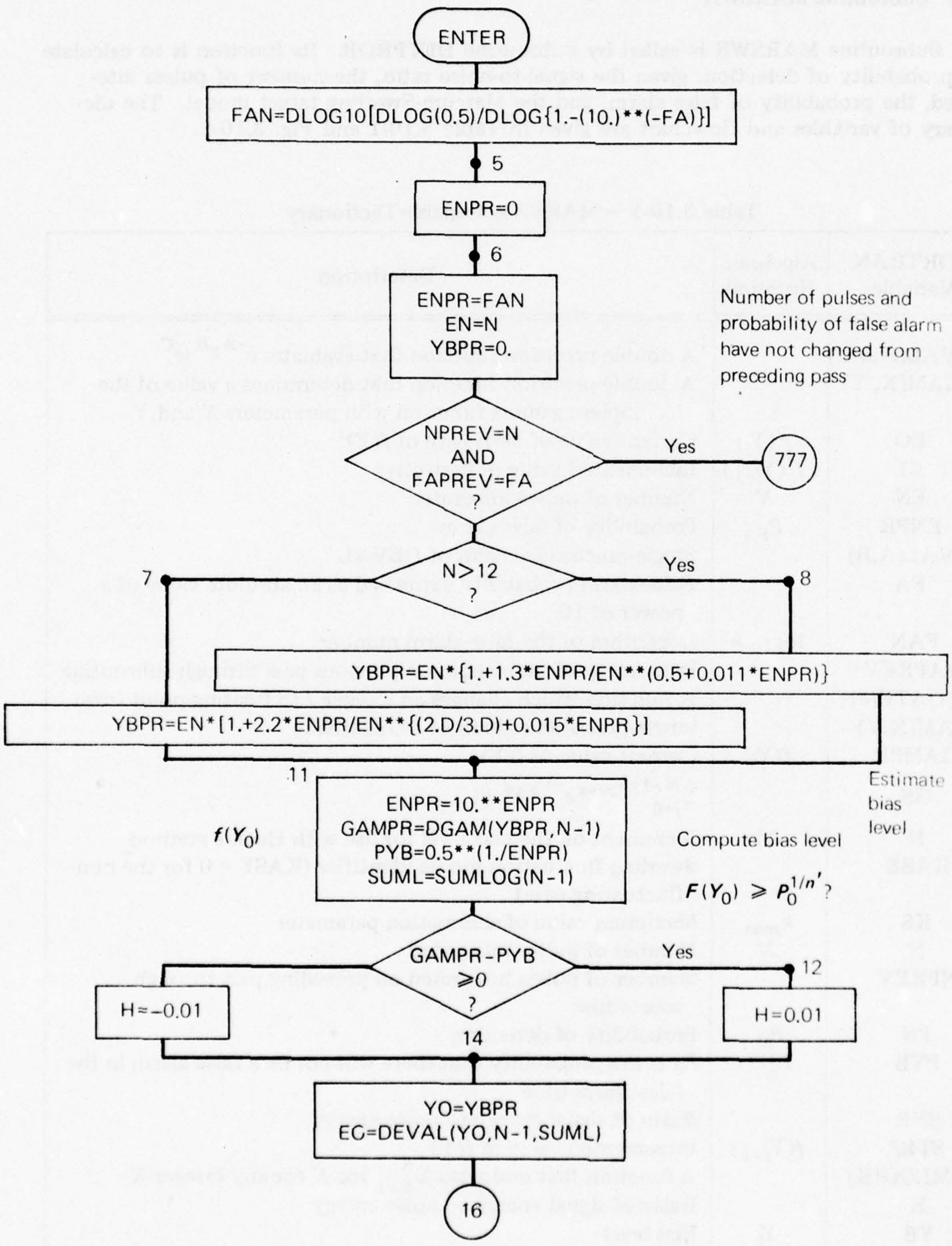


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart

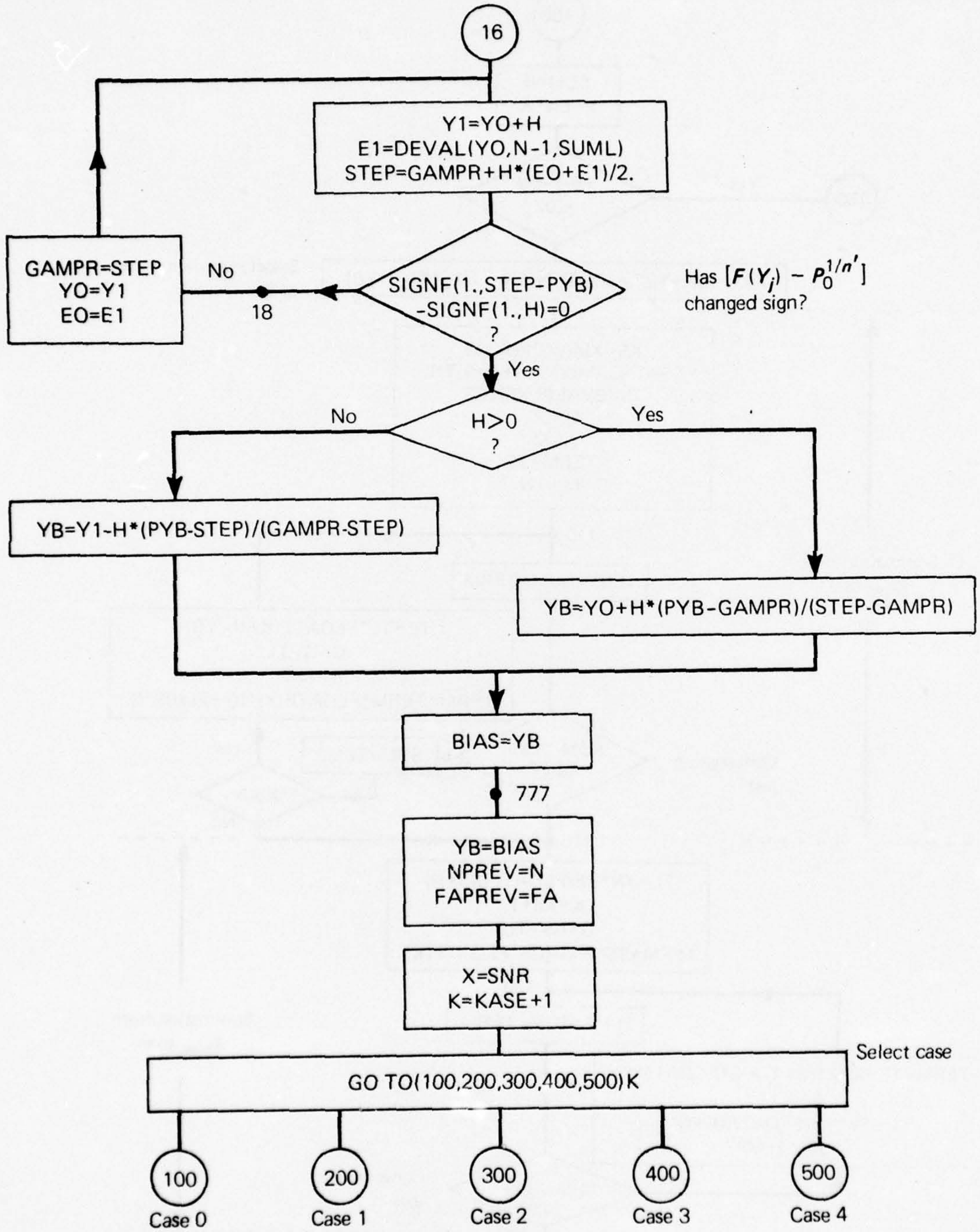


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

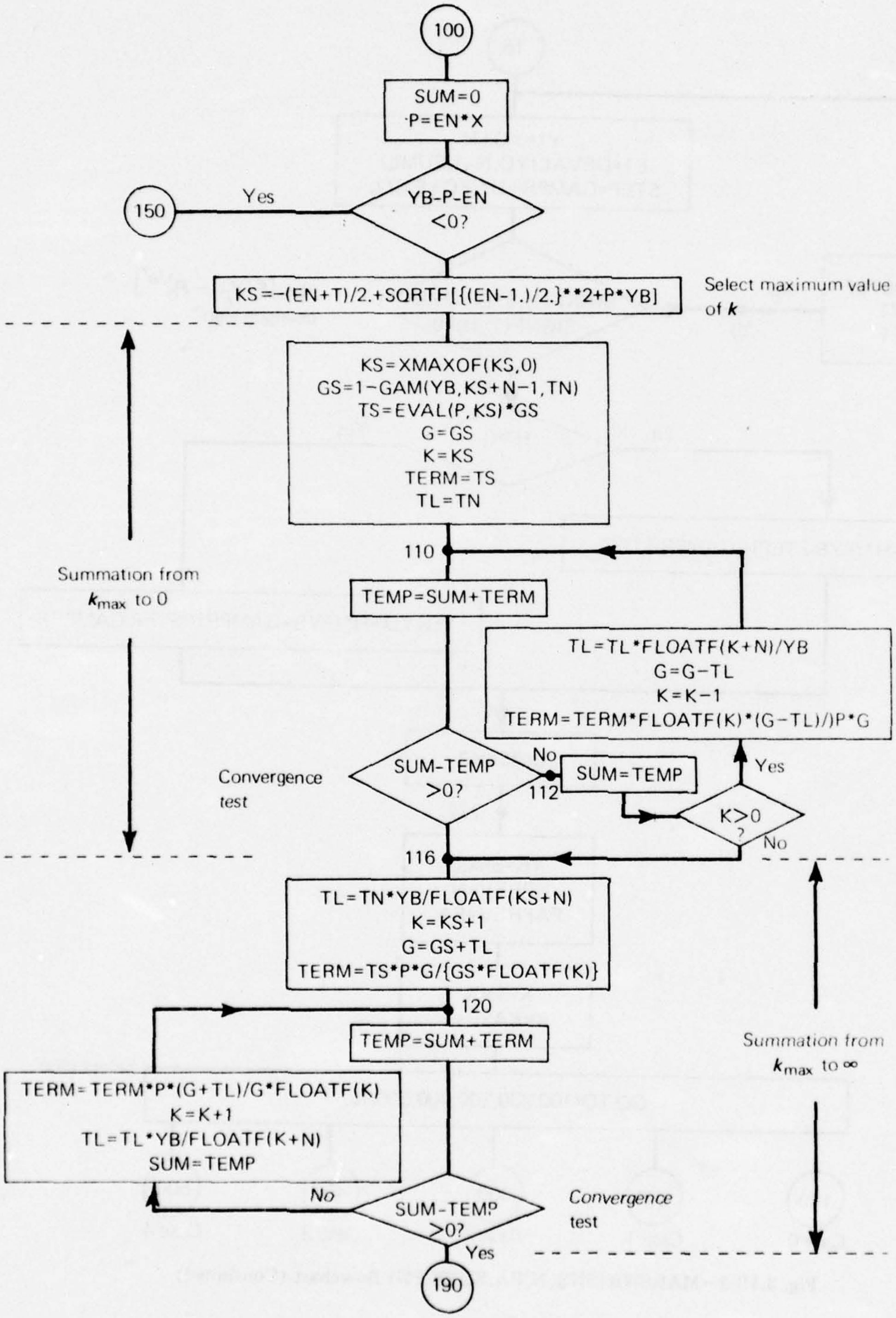


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

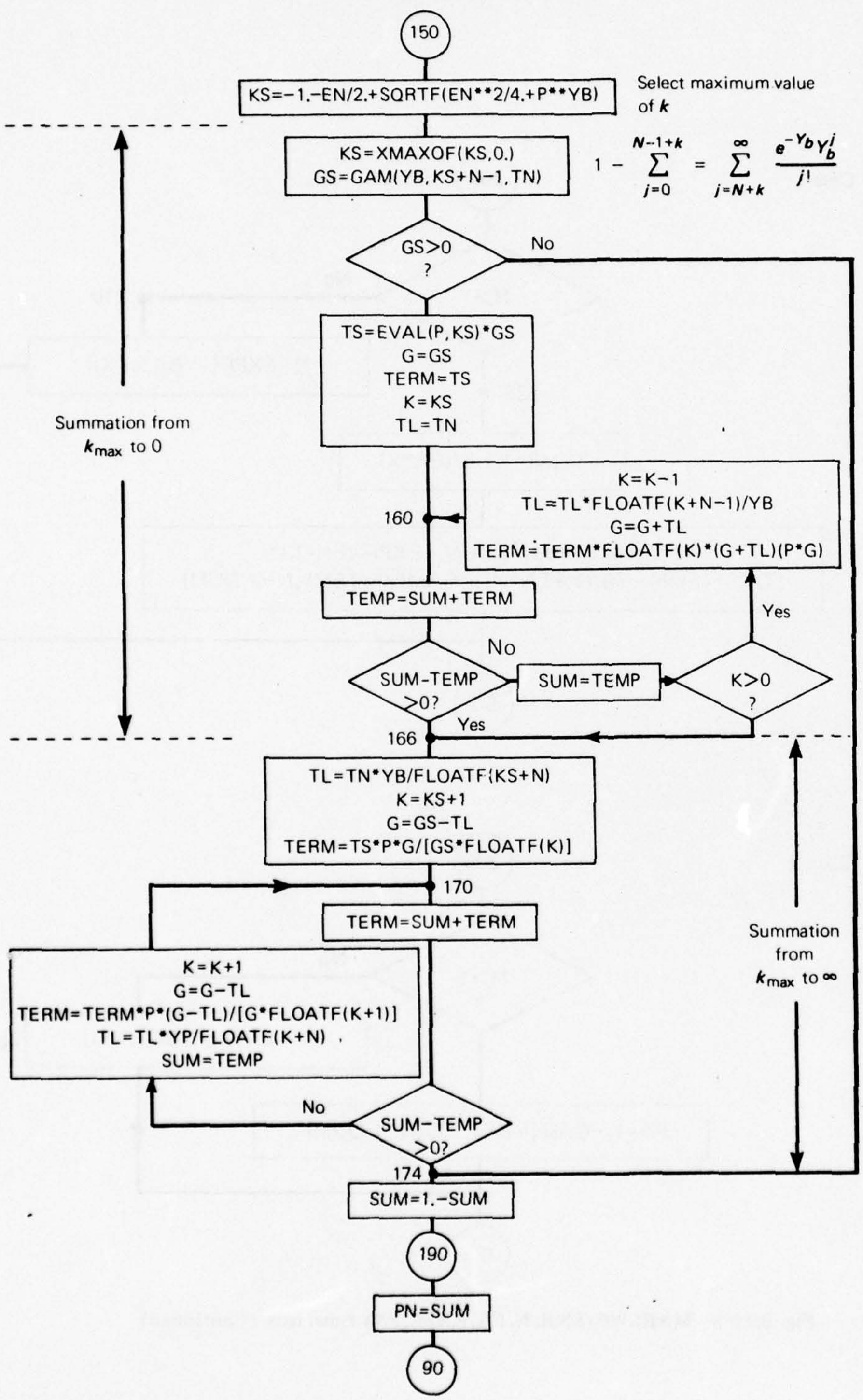


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

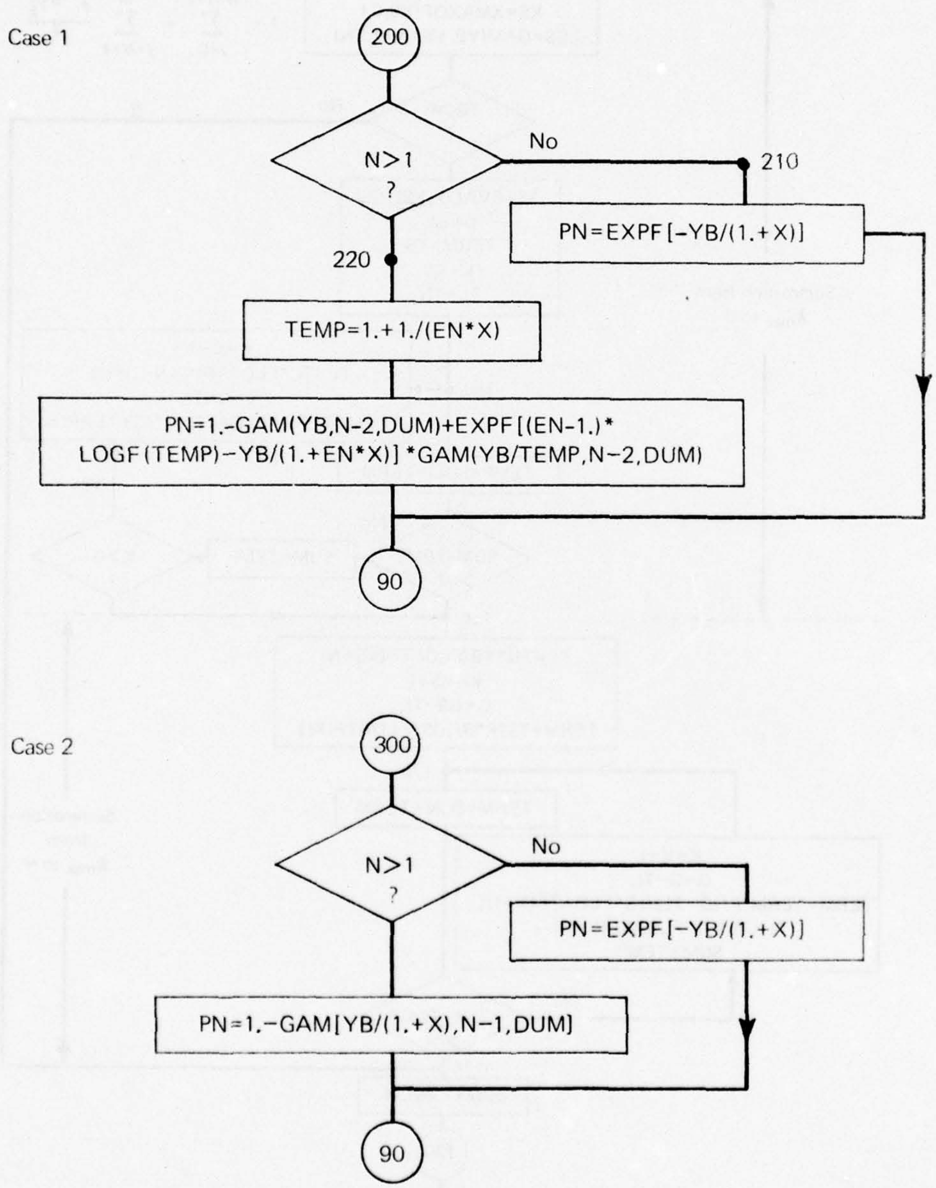


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

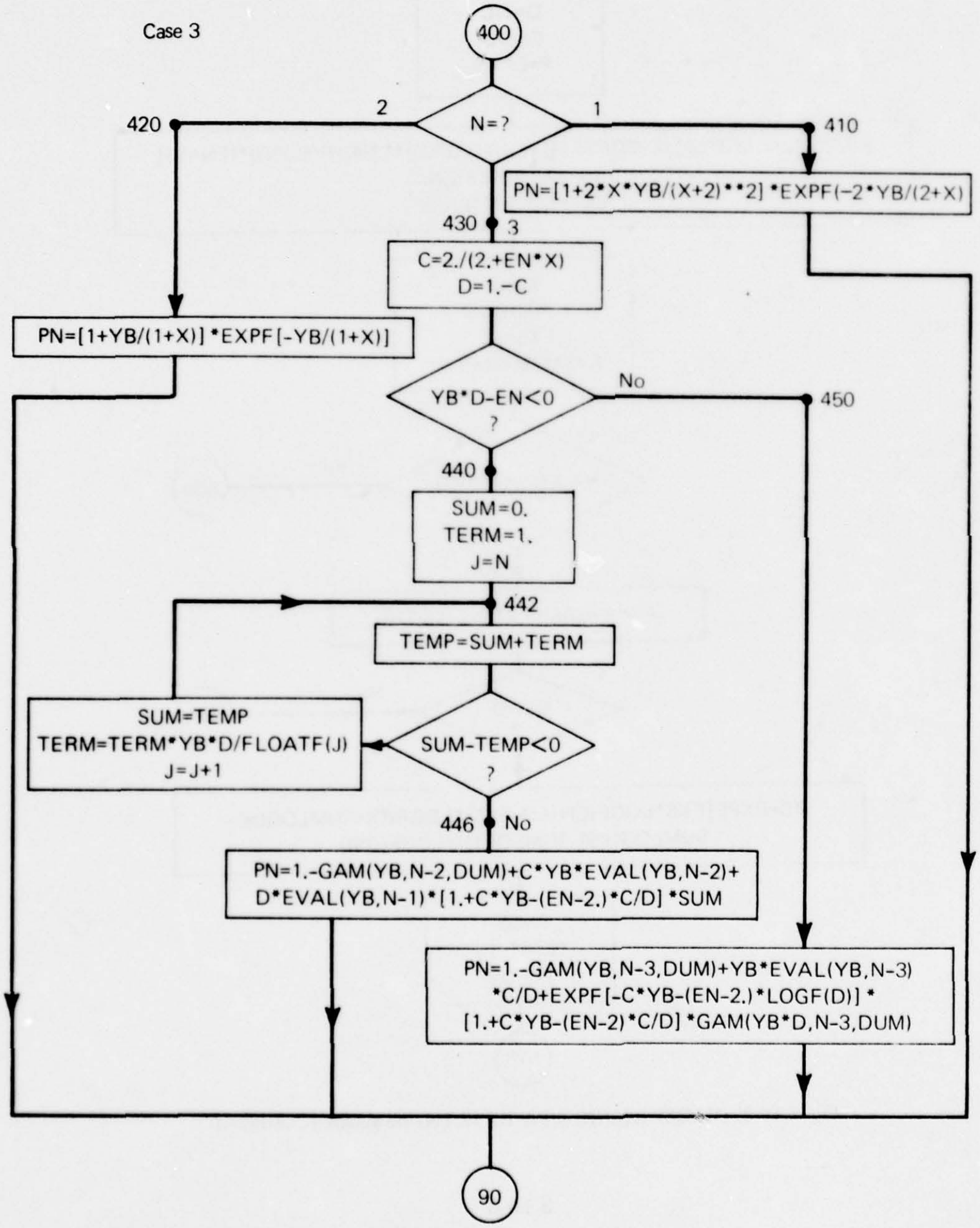


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

Case 4

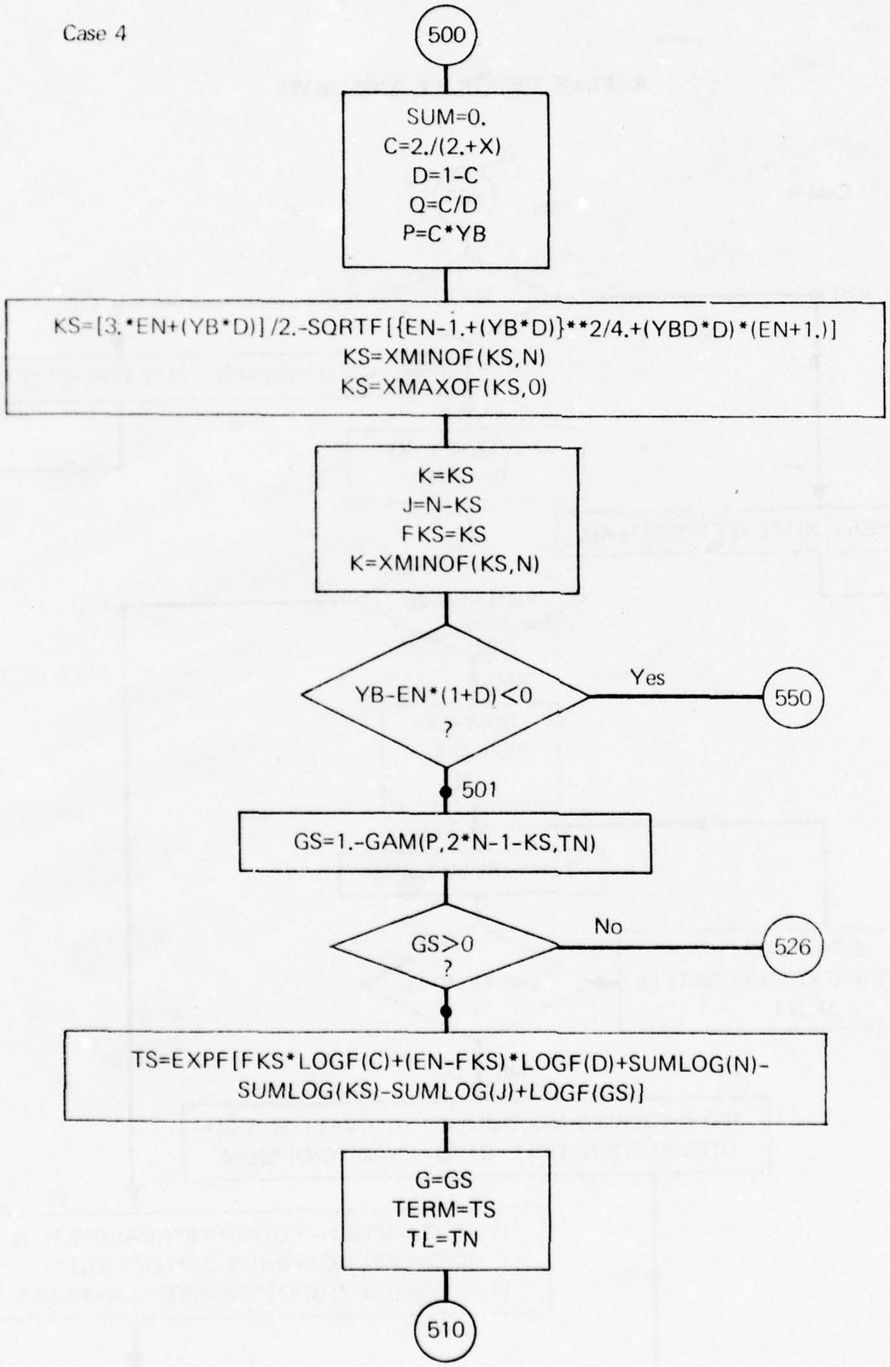


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

Case 4 (Continued)

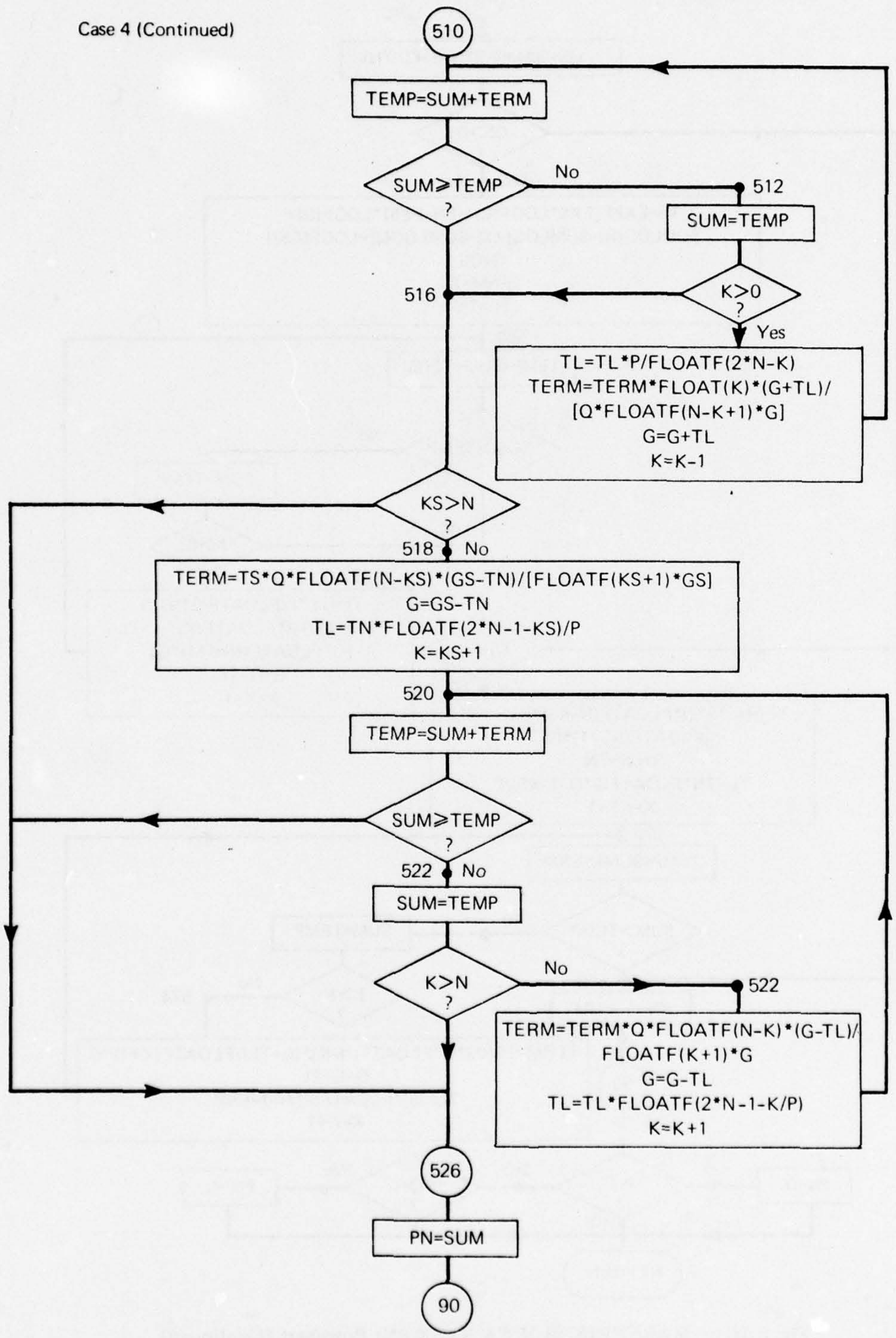


Fig. 3.10-2—MARSWR(SNR, N, FA, KASE, PN) flowchart (Continued)

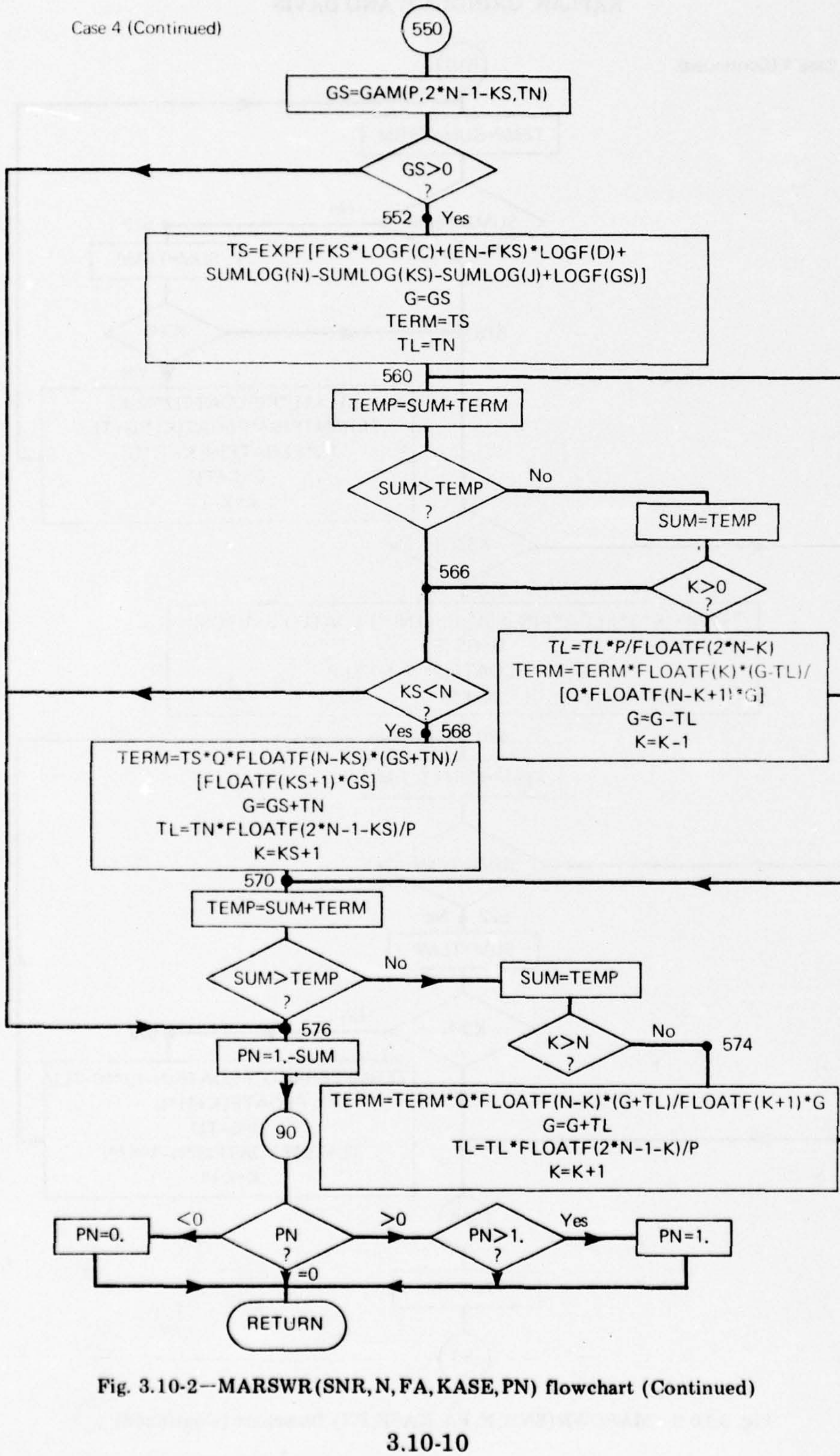


Fig. 3.10-2—MARSWR(SNR,N,FA,KASE,PN) flowchart (Continued)

The subroutine is entered with a value for the exponent of the false-alarm probability. This is initially converted to the logarithm of the false-alarm number (n') according to

$$\log_{10} n' = \log_{10}\left[\frac{\ln P_0}{\ln(1 - P_{FA})}\right]. \qquad (3.10\text{-}1)$$

For a pulse radar

$$n' = \frac{n}{N} \qquad (3.10\text{-}2)$$

and

n = the number of pulse lengths in the false-alarm time

N = the number of pulses integrated

P_0 = the probability of false alarm during the false-alarm time (taken to be 0.5)

P_{FA} = the probability of false alarm at each opportunity.

Having determined the false-alarm number, the next step in the process is to establish the bias level. This is derived for the square law of the combined detector and integrator. Marcum and Swerling [1] have shown that the square law permits an easy solution that differs little from the linear law. Consequently the bias level Y_b is calculated from

$$f(Y_b) = P_0^{1/n'} = \int_0^{Y_b} \frac{Y^{N-1} e^{-Y}}{(N-1)!}\, dY. \qquad (3.10\text{-}3)$$

This is accomplished through the use of an iterative procedure. An initial value for the bias level (Y_0) is calculated from equations that are fitted to curves developed by Marcum:

$$Y_0 = N\left[1 + \frac{2.2n'}{N^{(2/3+0.015n')}}\right] \quad (N > 12) \qquad (3.10\text{-}4)$$

and

$$Y_0 = N\left[1 + \frac{1.3n'}{N^{(0.5+0.011n')}}\right] \quad (N \leqslant 12). \qquad (3.10\text{-}5)$$

The integral of Eq. (3.10-3) is in the form of the incomplete gamma function, which is readily integrated by using the value Y_0 for Y_b. This value, $f(Y_0)$, is compared with $P_0^{1/n}$ or $f(Y_b)$ and, if the difference is greater than zero to eight significant digits, a fixed-step integration process (Huen's method) is initiated. This produces an improved value according to

$$f(Y_{J+1}) = f(Y_J) + \frac{\Delta Y}{2}\,[f'(Y_J) + f'(Y_{J+1})] \quad (J = 0, 1) \qquad (3.10\text{-}6)$$

where

$$f'(Y) = \frac{Y^{N-1}e^{-Y}}{(N-1)!} \tag{3.10-7}$$

and

$$Y_{J+1} = Y_J + \Delta Y . \tag{3.10-8}$$

A value of ±0.01 is currently being used for ΔY and has been found to provide sufficient accuracy. The integration is continued until $f(Y_j)$ is sufficiently close to $f(Y_b)$.

Having selected the bias level, the probability of detection is now determined for the appropriate Marcum-Swerling target case. The five cases are identified by number, with the nonfluctuating case being designated case 0. Each case will be considered in turn.

Case 0

The probability of detection for a target with a nonfluctuating cross section (case 0) is given by Marcum's Eq. (49),

$$P_N = \int_{Y_b}^{\infty} \left(\frac{Y}{Nx}\right)^{(N-1)/2} e^{-Y-Nx} I_{N-1}(2\sqrt{NxY})\, dY , \tag{3.10-9}$$

where x is the signal-to-noise ratio and I_N is the modified Bessel function of the first kind.

Fehlner [6] has shown that this integral can be expressed as an infinite summation as follows:

$$P_N = e^{-Nx} \sum_{k=0}^{\infty} \frac{(Nx)^k}{k!} \sum_{j=0}^{N-1+k} \frac{e^{-Y_b} Y_b^j}{j!} \tag{3.10-10}$$

or alternatively

$$P_N = 1 - e^{-Nx} \sum_{k=0}^{\infty} \frac{(Nx)^k}{k!} \sum_{j=N+k}^{\infty} \frac{e^{-Y_b} Y_b^j}{j!} . \tag{3.10-11}$$

It has been found that Eq. (3.10-10) converges faster than Eq. (3.10-11) if $Y_b > N(x+1)$. The summation is carried out within the program by first identifying the maximum term and then performing the summation about that point. For Eq. (3.10-10) the maximum term is associated with the value of k that maximizes

$$\frac{(Nx)^k}{k!} \frac{Y_b^{N-1+k}}{(N-1+k)!} . \tag{3.10-12}$$

Through the use of Stirling's approximation for the factorial, the value of k that produces a maximum is estimated to be

$$k_{\max} = -\frac{(N+1)}{2} + \left[\left(\frac{N-1}{2}\right)^2 + Y_b N x\right]^{1/2} . \qquad (3.10\text{-}13)$$

For Eq. (3.10-11) the maximum term is associated with the value of k that maximizes

$$\frac{(Nx)^k}{k!} \; \frac{Y_b^{N+k}}{(N+k)!} . \qquad (3.10\text{-}14)$$

This is found to be

$$k_{\max} = -\left(1 + \frac{N}{2}\right) + \left(\frac{N^2}{4} + Y_b N x\right)^{1/2} . \qquad (3.10\text{-}15)$$

Case 1

Swerling extended Marcum's square law results to cover fluctuating-target returns. Case 1 applies to targets that can be considered as several large independent fluctuating reflectors with approximately the same reflective cross section. The echo pulses are assumed to be constant on any given scan but are independent from scan to scan. The probability density function for the input signal-to-noise ratio is assumed to be

$$W(x, \bar{x}) = \frac{1}{\bar{x}} e^{-x/\bar{x}} \quad (x \geqslant 0) \qquad (3.10\text{-}16)$$

where $\bar{x}$ is the mean signal to noise ratio over all target fluctuation. The resulting formulas for the probability of detection are then given by

$$P_D = \exp\left(-\frac{Y_b}{1 + \bar{x}}\right) \quad (N = 1) \qquad (3.10\text{-}17)$$

and

$$P_D = 1 - \int_0^{Y_b} \frac{e^{-v} v^{N-2}}{(N-2)!} \, dv + \left(1 + \frac{1}{N\bar{x}}\right) \exp\left(-\frac{Y_b}{1 + \bar{x}}\right)$$

$$\times \int_0^{Y_b/(1+1/N\bar{x})} \frac{e^{-v} v^{N-2}}{(N-2)!} \, dv \quad (N > 1) . \qquad (3.10\text{-}18)$$

Both integrals in the above equation are in the form of the incomplete gamma function and are readily integrated through the use of the GAM function.

Case 2

Case 2 applies to the situation in which the fluctuations are independent from pulse to pulse and the probability density function for the input signal-to-noise ratio is given by Eq. (97). The resulting formulas for the probability of detection are given by

$$P_D = 1 - \int_0^{Y_b/(1+\bar{x})} \frac{e^{-v} v^{N-1}}{(N-1)!}\, dv . \tag{3.10-19}$$

For $N = 1$ this reduces to

$$P_D = \exp\left(-\frac{Y_b}{1+\bar{x}}\right). \tag{3.10-20}$$

The integral in Eq. (3.10-19) is in the form of the incomplete gamma function, which again can be readily evaluated through the use of the GAM function.

Case 3

Case 3 applies to targets that can be represented as one large reflector together with other small reflectors or as one large reflector subject to fairly small changes in orientation. This type of target exhibits a probability density for the input signal-to-noise ratio according to

$$W(x, \bar{x}) = \frac{4x}{(\bar{x})^2} e^{-2x/\bar{x}} \qquad (x \geqslant 0) . \tag{3.10-21}$$

For case 3 the fluctuations are considered to be independent from scan to scan. The resulting formula for the probability of detection is given by

$$\begin{aligned} P_D = 1 &+ \frac{C^2}{(C-1)^{N-2}} \int_0^{Y_b} v e^{-vC} \int_0^{-v/(C-1)} \frac{e^{-u} u^{N-2}}{(N-2)!}\, dv\, du \\ &+ \frac{(N-2)C^2}{(C-1)^{N-1}} \int_0^{Y_b} e^{-vC} \int_0^{-v/(C-1)} \frac{e^{-u} u^{N-2}}{(N-2)!}\, dv\, du \\ &- (N-1)C^2 \int_0^{Y_b} \frac{e^{-u} u^{N-1}}{(N-1)!}\, du \end{aligned} \tag{3.10-22}$$

where

$$C = \frac{1}{1 + N\bar{x}/2}. \tag{3.10-23}$$

Fehlner [6] has shown that Eq. (3.10-22) can be expressed as follows:

$$P_D = \frac{CY_b^{N-1}e^{-Y_b}}{(N-2)!} + \sum_{j=0}^{N-2} \frac{e^{-Y_b}Y_b^j}{j!}$$

$$+ \frac{e^{-CY_b}}{(1-C)^{N-2}}\left[1 - \frac{(N-2)C}{(1-C)} + CY_b\right]\left\{1 - \sum_{j=0}^{N-2} \frac{e^{-Y_b(1-C)}[Y_b(1-C)]^j}{j!}\right\} \quad (3.10\text{-}24)$$

For $N = 1$ this reduces to

$$P_D = \left[1 + \frac{2\bar{x}Y_b}{(\bar{x}+2)^2}\right] \exp\left(-\frac{2Y_b}{2+\bar{x}}\right), \quad (3.10\text{-}25)$$

and for $N = 2$,

$$P_D = \left(1 + \frac{Y_b}{1+\bar{x}}\right) \exp\left(-\frac{Y_b}{1+\bar{x}}\right). \quad (3.10\text{-}26)$$

For $N > 2$, Eq. (3.10-24) is readily evaluated with the GAM function.

Case 4

Case 4 applies to targets that exhibit the same probability density function as those considered in case 3 but with the fluctuations considered to be independent on a pulse-to-pulse basis.

The resulting formula for the probability of detection is given by

$$P_D = 1 + \frac{1}{(g-1)^N} \sum_{k=0}^{N} \left(\frac{\bar{x}}{2}\right)^{-k} \frac{N!}{k!(N-k)!} \int_0^{Y_b g} \frac{e^{-u}u^{(2N-1-k)}}{(2N-1-k)!}\, du \quad (3.10\text{-}27)$$

where

$$g = \frac{1}{1 + \bar{x}/2}. \quad (3.10\text{-}28)$$

Equation (3.10-27) can be expressed as

$$P_D = 1 - g^N \sum_{k=0}^{N} \frac{N!}{k!(N-k)!}\left(\frac{1-g}{g}\right)^{N-k} \sum_{j=2N-k}^{\infty} \frac{e^{-gY_b}(gY_b)^j}{j!}. \quad (3.10\text{-}29)$$

After the probability-of-detection calculation is completed, control of the program is returned to DETPROB.

3.11 Subroutine MATCH

Subroutine MATCH is called by the SURSEM Executive Routine. For each scan mode the subroutine calculates the time at which the mode first scans the target after it has come within the instrumented range.

This is accomplished as follows: The components of vectors **A** and **B** (see Fig. 3.11-1) are first calculated from the input initial and final position of the target and the coordinates of the ship. The distance d is then calculated from

$$d^2 = \left|\frac{\mathbf{A}\cdot\mathbf{B}}{B}\right| - (A^2 - R_{ins}^2) \qquad (3.11\text{-}1)$$

where R_{ins} is the instrumented range. The ratio $x/|\mathbf{B}|$ is now found:

$$\frac{x}{|\mathbf{B}|} = \frac{\text{distance from initial point to a range of } R_{ins}}{\text{distance from initial point to final point}} \qquad (3.11\text{-}2)$$

and is given by

$$\frac{x}{|\mathbf{B}|} = -\frac{1}{|\mathbf{B}|}\left(\frac{\mathbf{A}\cdot\mathbf{B}}{|\mathbf{B}|} + d\right). \qquad (3.11\text{-}3)$$

This ratio is used to determine the time t_R required for the target to travel from its initial point to the point where it comes within instrumented range, that is,

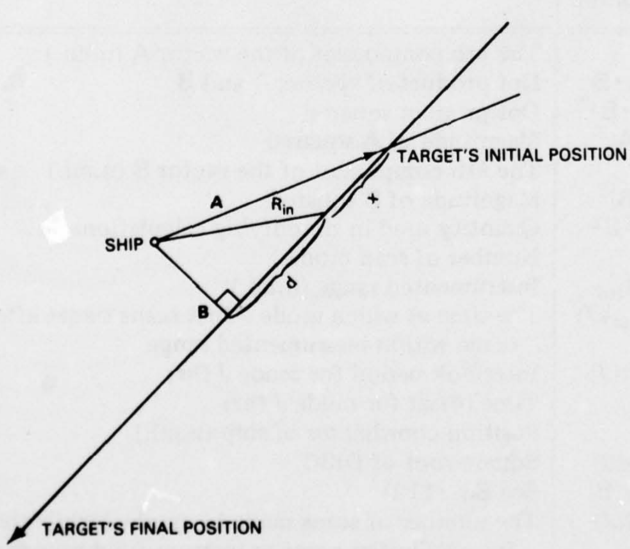


Fig. 3.11-1 —Target-ship geometry

$$t_R = t_i + \frac{x}{|\mathbf{B}|}(t_f - t_i) \tag{3.11-4}$$

where t_i and t_f represent the initial and final times of the target's trajectory, respectively. The quantity

$$N(J) = \frac{t_R - t_i(J)}{\Delta t(J)} \tag{3.11-5}$$

is found and truncated to the next lowest integer. This represents the number of scans made by mode J during the time interval in which the target is covering the distance from its initial point to a range of R_{ins}. The time at which mode J first scans the target after it has come within the radar's instrumented range can now be determined.

$$t_{scan}(J) = t_i(J) + [N(J) + 1]\,\Delta t(J) \tag{3.11-6}$$

where $t_i(J)$ is the starting time for mode J. This is determined by assuming that the radar is initially scanning in mode 1 and pointing at the target. Time offsets between the modes are provided by inputs, and the quantity $\Delta t(J)$ represents the interlook period for mode J.

Table 3.11-2 is a dictionary of the variables used in Subroutine MATCH. The flow-chart is shown in Fig. 3.11-3.

Table 3.11-2 — MATCH Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
A(K)		The kth component of the vector **A** (n.mi.)
ADOTB	$\mathbf{A}\cdot\mathbf{B}$	Dot product of vectors **A** and **B**
ADOTB2	$(\mathbf{A}\cdot\mathbf{B})^2$	Dot product squared
AMAG2	$\|\mathbf{A}\|^2$	Magnitude of **A** squared
B(K)		The kth component of the vector **B** (n.mi.)
BMAG2	$\|\mathbf{B}\|^2$	Magnitude of **B** squared
DISC	d^2B^2	Quantity used in simplifying calculations
NSCAN		Number of scan modes
RMODE(I,7)	R_{ins}	Instrumented range (n.mi.)
RMODE(I,9)	$t_{scan}(J)$	The time at which mode J first scans target after the target has come within instrumented range
RMODE(J,5)	$\Delta t(J)$	Interlook period for mode J (hr)
RMODE(J,6)		Time offset for mode J (hr)
SHIP(K)		Position coordinates of ship (n.mi.)
TERM	$\|dB\|$	Square root of DISC
UMINUS	$x/\|\mathbf{B}\|$	See Eq. (112)
XN	$N(J)$	The number of scans made by mode J while the target is going from its initial point to instrumented range
XSCAN	$t_{scan}(J)$	Time at which mode J first scans the target after it has come within instrumented range (hr)
XYZF(1,K)		Final position coordinates of target 1 (n.mi.)
XYZI(1,K)		Initial position coordinates of target 1 (n.mi.)

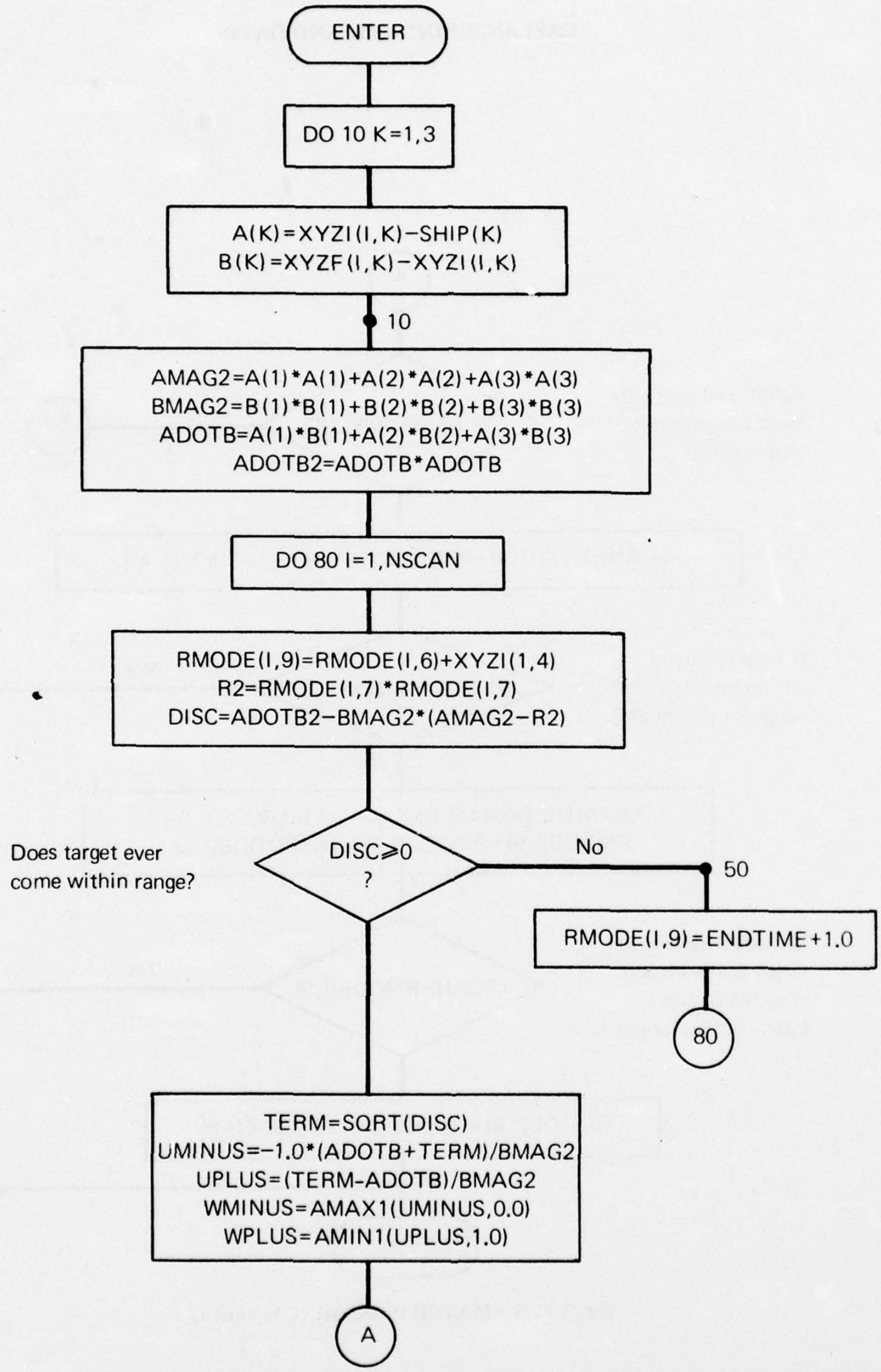


Fig. 3.11-3—MATCH flowchart

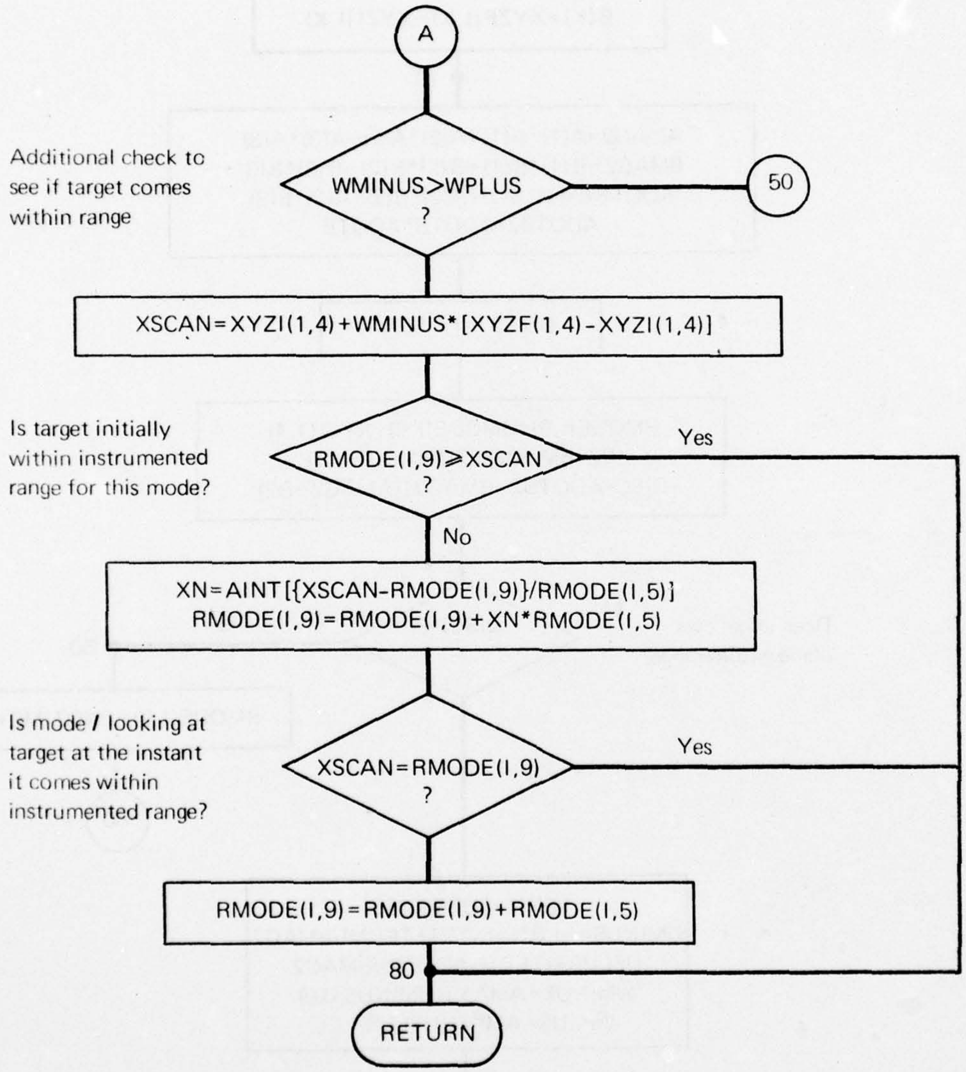


Fig. 3.11-3—MATCH flowchart (Continued)

3.12 Subroutine MULPTH

Subroutine MULPTH is called by Subroutines DETPROB and JAM. Its purpose is to calculate the pattern-propagation factor (FAC) for a specified target (ITAR).

For computational purposes the atmosphere is divided into three regions: the interference region, the intermediate region, and the diffraction region (see Fig. 3.12-1). The propagation factor is determined only for targets above the horizon—that is, in the interference region and part of the intermediate region. This can be readily accomplished for targets in the interference region and diffraction regions, but there are no numerical methods that are easily applicable to field-strength determination in the intermediate region. Consequently the method employed to find the pattern-propagation factor for targets in the intermediate region is an interpolation procedure that interpolates between the interference region and the diffraction region. In other words, the field strength is determined for points in the interference and diffraction regions at the altitude of the target under consideration, and this information is then used to determine the field strength in the intermediate region by a curve-fitting process. Details of the analytical process are outlined below.

The initial step in the calculation process, which is carried out for each new frequency mode, is the determination of the complex dielectric constant, given by

$$\epsilon_c = \epsilon_1 - i60\lambda\sigma \qquad (3.12\text{-}1)$$

where ϵ_1 is the ordinary dielectric constant, λ is the wavelength, and σ is the conductivity. Values of ϵ_1 and σ as a function of frequency are given in Refs. 7 and 8. The values used in MULPTH are given in Table 3.12-2. A linear interpolation process is used for intermediate values.

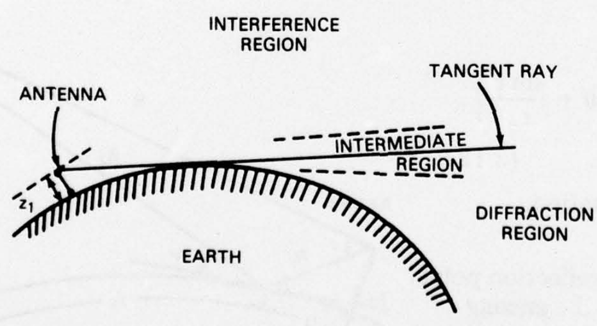


Fig. 3.12-1—Interference, intermediate, and diffraction regions

Table 3.12-2—Values of Frequency, Dielectric Constant, and Conductivity Used in MULPTH

f (MHz)	ϵ_1	σ (mhos/m)
<1500	80	4.3
1500 to 3000	$80 - [0.00733(f - 1500)]$	$4.3 + [0.00148(f - 1500)]$
3000 to 10 000	$69 - [0.0005714(f - 3000)]$	$6.52 + [0.001354(f - 3000)]$

Fig. 3.10-2—MARSWR(SNR,N,FA,KASE,PN) flowchart (Continued)

The next step in the calculation is the determination of factors that will b used in calculating the value of the pattern-propagation factor for a target in the diffraction region. The propagation factor in the diffraction region is given by (see Ref. 8, page 141)

$$F = 2\sqrt{\pi X}\, e^{-C_1 x}\, |U_1(Z_1) U_1(Z_2)| . \tag{3.12-2}$$

X is the target ground range in natural units,

$$X = \frac{r}{L} , \tag{3.12-3}$$

where L is the natural unit of range,

$$L = 2\left(\frac{4k_0}{a^2}\right)^{-1/3} , \tag{3.12-4}$$

a is the Earth's effective radius, and k_0 is the radar wave number multiplied by the index of refraction at the Earth's surface. Only the factors that are target independent—namely, C_1 and $U_1(Z_1)$—are calculated on the initial pass for a new frequency mode. The term C_1 is the imaginary part of the parameter A_1, which is also used in evaluating $U_1(Z_1)$ in Subroutine UFUN. The parameter A_1 is given by (Ref. 8, page 111)

$$A_1 = 2.3381 \exp\left(\frac{2\pi i}{3}\right) + \frac{1}{Hp} \tag{3.12-5}$$

where H is the natural unit of height,

$$H = \left(\frac{2k_0^2}{a}\right)^{-1/3} , \tag{3.12-6}$$

p is given by

$$p = ik_0(\epsilon_c - 1)^{1/2}\left(\cos\theta + \frac{\sin\theta}{\epsilon_c}\right) , \tag{3.12-7}$$

and θ is the linear polarization.

The location of the reflection point and the determination of the grazing angle is the next step in the calculation process. The ground range of the reflection point (r_1) is a function of target height (h_2), antenna height (h_1), and target ground range (r) (see Fig. 3.12-3). This relationship is expressed by

Fig. 3.12-3—Geometrical parameters

$$2r_1^3 - 3rr_1^2 + [r^2 - 2a(H_1 + H_2)]r_1 + 2aH_1r = 0 \tag{3.12-8}$$

where

$$H_i = h_i\left(1 - \frac{h_i}{a}\right) \quad (i = 1, 2)\,. \tag{3.12-9}$$

Equation (3.12-8) has the solution

$$r_1 = \frac{r}{2} + \mathrm{sgn}\,(H_1 - H_2)P\cos\left(\frac{\phi + \pi}{3}\right) \tag{3.12-10}$$

where

$$P = \left\{\frac{4}{3}\left[a(H_1 + H_2) + \left(\frac{r}{2}\right)^2\right]\right\}^{1/2} \tag{3.12-11}$$

and

$$\phi = \arccos\left(\frac{2ar|H_1 - H_2|}{P^3}\right). \tag{3.12-12}$$

The grazing angle is then found from the approximation

$$\tan\psi \approx \frac{h_1}{r_1}\left(1 - \frac{h_1}{a}\right) - \frac{r_1}{2a}\,. \tag{3.12-13}$$

If the grazing angle is found to be less than or equal to zero, the pattern-propagation factor is set equal to 1×10^{-20} and control of the program is returned to DETPROB. If ψ is greater than zero, the divergence factor D and the path-length difference ΔR are calculated according to

$$D = \left\{\left(1 - \frac{h_1 + h_2}{a}\right)\left[\frac{(R_1 + R_2)^2\, a\sin\psi\cos\psi}{(R_1 + R_2)a\sin\psi + 2R_1R_2}\right]\left(\frac{1}{r}\right)\right\}^{1/2} \tag{3.12-14}$$

and

$$\Delta R = \frac{(R_1 + R_2)G}{1 + (1 - G)^{1/2}} \tag{3.12-15}$$

where

$$G = \left(\frac{2\sin\psi}{R_1 + R_2}\right)^2 R_1R_2 \tag{3.12-16}$$

and R_1, R_2 are slant ranges from the antenna and target to the reflection point, respectively.

Equation (3.12-14) is a simplified version of Eq. (16) in Ref. 8, page 406, and Eq. (3.12-15) is readily derived from basic trigonometric relationships.

The path-length difference is now compared to $\lambda/4$ since this is the generally accepted limit of validity for the analytical method applied to the interference region. If $\Delta R < \lambda/4$, the target is assumed to be in the intermediate region, and the preliminary step in the interpolation process is carried out. This consists of finding the location of a point that has the altitude of the target under consideration but with a path-length difference of $\lambda/4$. The method used is discussed under Subroutine RNGCEN. This information is used in the interpolation scheme for determining the pattern-propagation factor in the intermediate region.

The next step in the calculation process is the computation of the complex reflection coefficient. The reflection coefficient is related to the linear polarization, and the equations for the horizontal polarization and vertical polarization reflection coefficients are given by

$$\Gamma_v = \rho_v e^{-i\phi_v} = \frac{\epsilon_c \sin\psi - (\epsilon_c - \cos^2\psi)^{1/2}}{\epsilon_c \sin\psi + (\epsilon_c - \cos^2\psi)^{1/2}} \tag{3.12-17}$$

and

$$\Gamma_h = \rho_h e^{-i\phi_h} = \frac{\sin\psi - (\epsilon_c - \cos^2\psi)^{1/2}}{\sin\psi + (\epsilon_c - \cos^2\psi)^{1/2}} \tag{3.12-18}$$

where ρ is the intrinsic reflection coefficient and ϕ is the phase change for seawater. For other than vertical or horizontal polarization the reflection coefficient is given as the vector sum of the horizontal and vertical components:

$$\Gamma = \rho_0 e^{-i\phi} = \left[(\Gamma_h \cos\theta)^2 + (\Gamma_v \sin\theta)^2\right]^{1/2} \tag{3.12-19}$$

The two remaining parameters that contribute to the pattern-propagation factor are the roughness factor and the directional field strength ratio. The roughness factor is calculated from a formula found in Ref. 7:

$$r_s = \exp\left[-2\left(\frac{2 - H\sin\psi}{\lambda}\right)\right]^2 . \tag{3.12-20}$$

Equation (3.12-20) was developed under the assumption that the sea surface has a Gaussian height distribution with standard deviation H_s. For a sea surface with approximately sinusoidal waves and amplitude a.

$$H_s = \frac{a}{2\sqrt{2}} . \tag{3.12-21}$$

The field-strength ratios $f(\theta_1)$ and $f(\theta_2)$ for the direct and reflected rays are computed in Subroutine GAIN.

It is now possible to compute the pattern-propagation factor for the point in question according to

$$F = \left| 1 + \frac{f(\theta_2)}{f(\theta_1)} \rho_0 D r_s \exp(-i[k_0 \Delta R + \phi]) \right| \tag{3.12-22}$$

The point in question will be either the target position if the target is in the interference region or the point corresponding to a path-length difference of $\lambda/4$ if the target is above the horizon and in the intermediate region.

For targets in the interference region the calculation process is now complete, and control of the program is returned to DETPROB.

For targets in the intermediate region additional computation is required. The pattern-propagation factor must be found for a point in the diffraction region that will be used as the lower bound in the interpolation procedure. A point that is twice the horizon distance from the antenna is chosen as being representative of the diffraction region. Subroutine UFUN is now called to determine the value of the parameter $U_1(Z_2)$ for this point, and the pattern-propagation factor is calculated according to Eq. (3.12-2). The upper bound for the interpolation procedure is the value of pattern-propagation factor that was calculated for the point with a path-length difference of $\lambda/4$ [Eq. (3.12-22)]. The lower and upper bound values of the pattern-propagation factor are presented to Subroutine INTER, which carries out the interpolation process to determine the pattern-propagation factor at the target. This completes the calculation process, and control of the program is returned to Subroutine DETPROB.

A dictionary of the variables used in Subroutine MULPTH is given in Table 3.12-4. Figures 3.12-5 and 3.12-6 are macro and micro flowcharts, respectively.

Table 3.12-4 — MULPTH Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
CA	A_1	Function of complex dielectric constant and polarization used in UFUN
CGAM	Γ	Reflection coefficient
CGAMH	Γ_H	Horizontal polarization reflection coefficient
CGAMV	Γ_V	Vertical polarization reflection coefficient
CTEMY	ϵ_c	Complex dielectric coefficient
CU1	$U_1(Z_1)$	Parameter used in evaluating F (see UFUN)
DISP	D	Divergence factor
EPSI	ϵ_1	Ordinary dielectric constant

AD-A039 841   NAVAL RESEARCH LAB WASHINGTON D C   F/G 17/9
SURVEILLANCE RADAR SYSTEMS EVALUATION MODEL (SURSEM) HANDBOOK.(U)
JAN 77 D J KAPLAN, A GRINDLAY, L DAVIS
UNCLASSIFIED   NRL-8037   NL

2 OF 2
ADA039 841

END
DATE
FILMED
6-77

Table 3.12-4 — MULPTH Variable Dictionary (Continued)

FORTRAN Variable	Algebraic Notation	Description
FAC	F	Pattern-propagation factor
FAC1	F	Pattern-propagation factor
GD	$f(\theta_1)$	Field strength ratio for direct ray
GRRAD	r_1	Ground range from radar to reflection point (m)
GRRT	r	Ground range to target (m)
GRTAR	r_2	Ground range from target to reflection point (m)
GV	$f(\theta_2)$	Field strength ratio for reflected ray
H	H	Natural unit of height $= (2k_0^2/a)^{-1/3}$
HR	h_1	Height of antenna (m)
HT	h_2	Height of target (m)
ITAR		Target under consideration
ISWIT		Indicator takes on values 0, 1, where 0 = initial pass or new mode, 1 = same mode as preceding pass
L	L	Natural unit of length $= 2(k_0 4/a^2)^{-1/3}$
PHIREF	ϕ	Reflection coefficient phase angle (rad)
POLRZ	θ	Linear polarization (deg) (0° = horizontal, 90° = vertical)
PTHDIF	ΔR	Path-length difference (m)
RE	a	4/3 radius of the Earth (m)
RHOREF	ρ_0	Intrinsic reflection coefficient
SIG1	σ	Conductivity (mhos/m)
SRRAD	R_1	Slant range from radar to reflection point (m)
SRTAR	R_2	Slant range from target to reflection point (m)
TANPSI	$\tan \psi$	Tangent of grazing angle
WVL	λ	Wavelength (m)
XFRE		Frequency of the next mode to be considered (MHz)
XIMCA		Imaginary part of CA
XKPAR	k	Wave number $= 2\pi/\lambda$
XKZERO	k_0	kn_0
XMUR	r_s	Roughness factor
XNAT	r/L	Ground range to target in natural units
XNAT1		Range corresponding to $\Delta R = \lambda/4$ in natural units
XNAT2	$2r_H/L$	Twice the horizon range in natural units
XNZERO	n_0	Index of refraction at the Earth's surface
Z1	Z_1	Antenna height in natural units
Z2	Z_2	Target height in natural units

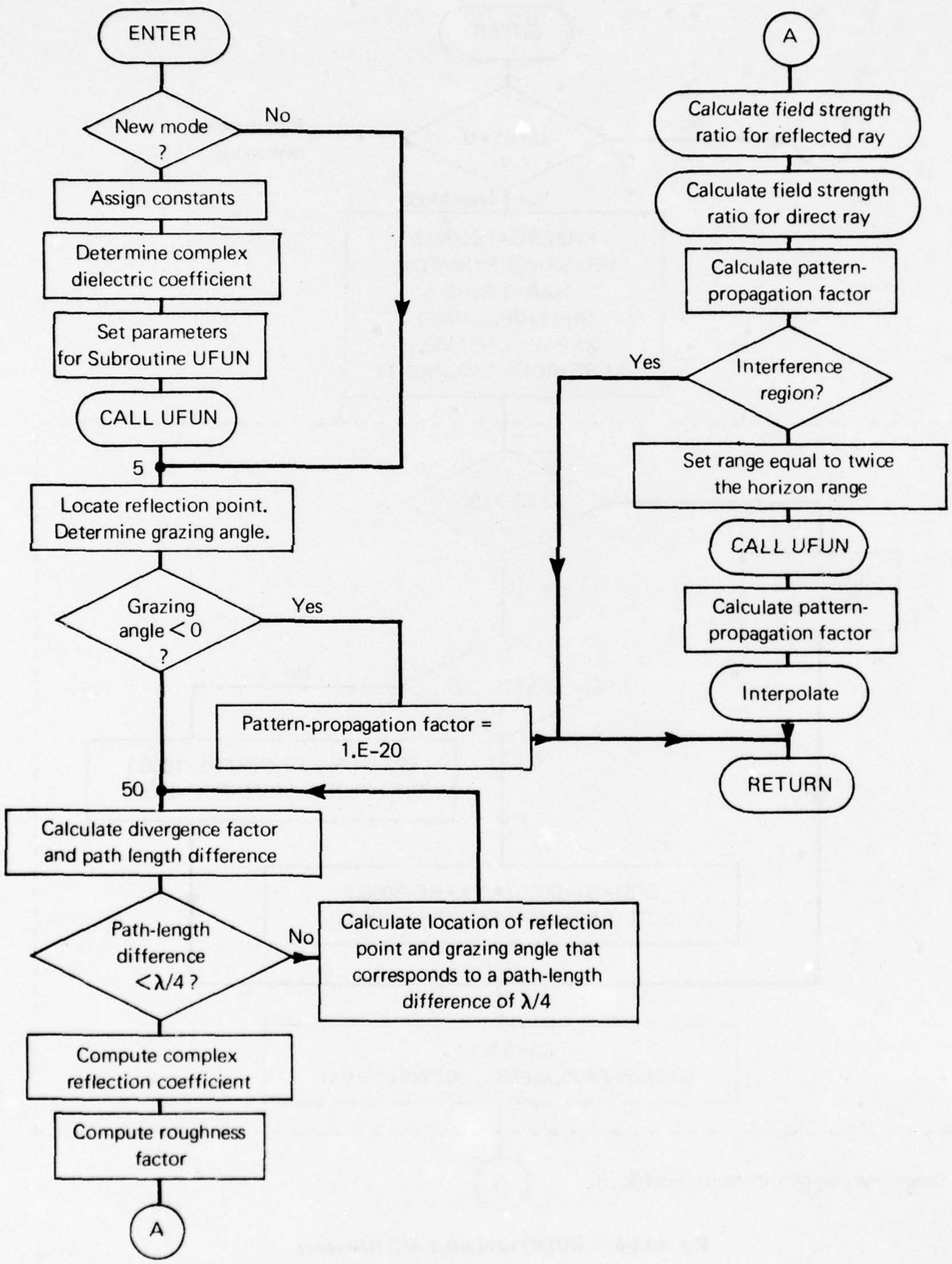


Fig. 3.12-5—MULPTH macro flowchart

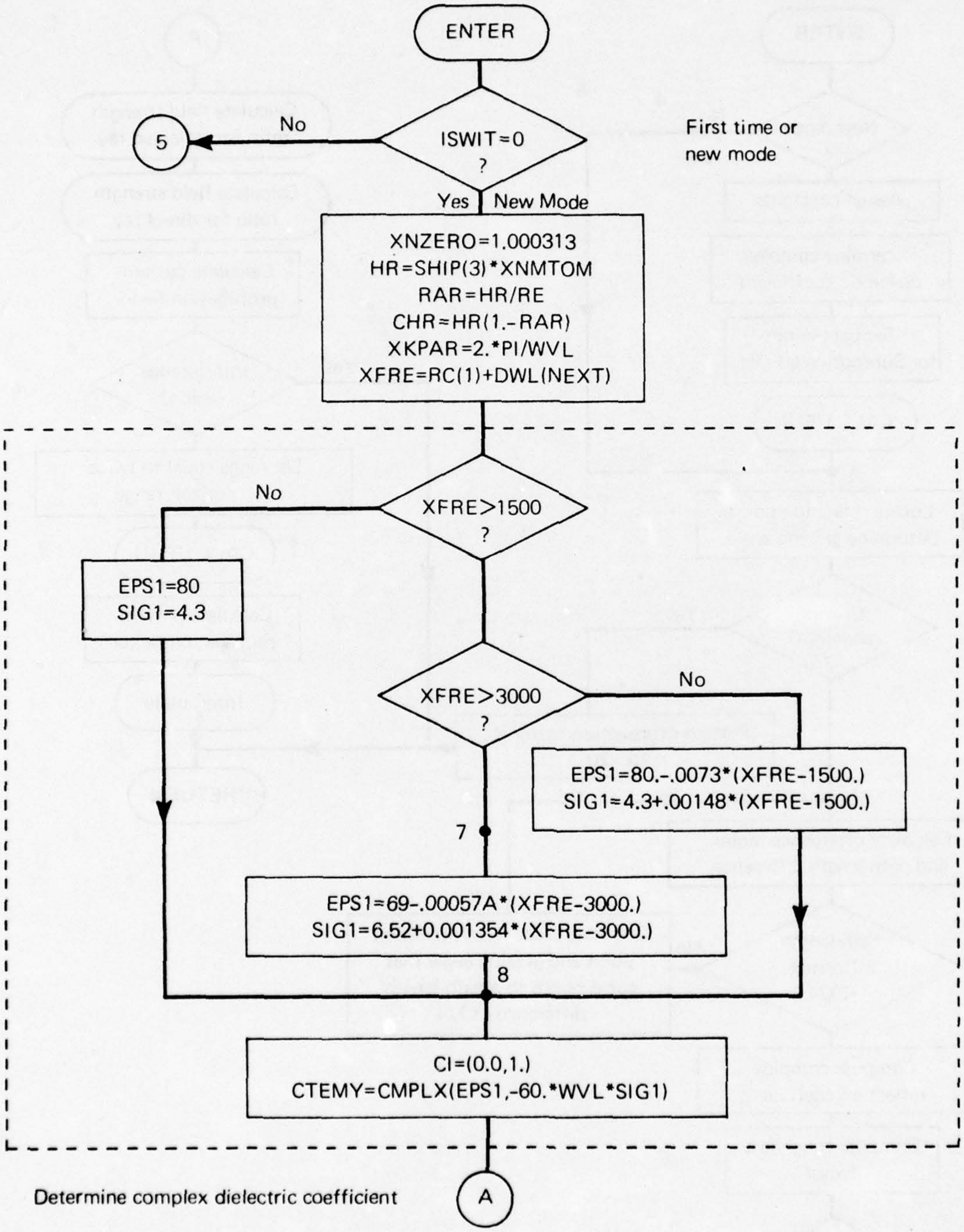


Fig. 3.12-6 — MULPTH(ITAR, FAC) flowchart

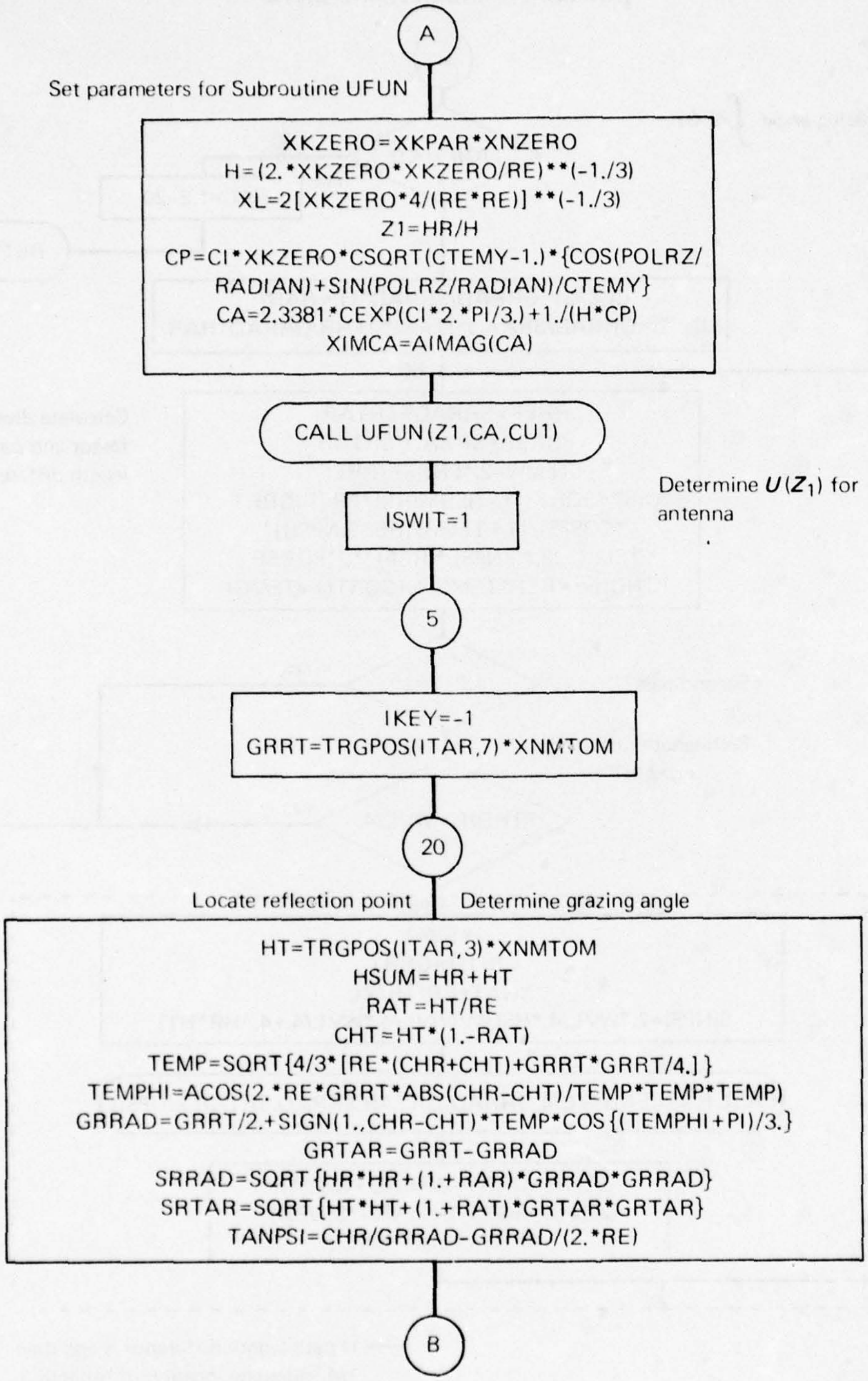


Fig. 3.12-6—MULPTH(ITAR,FAC) flowchart (Continued)

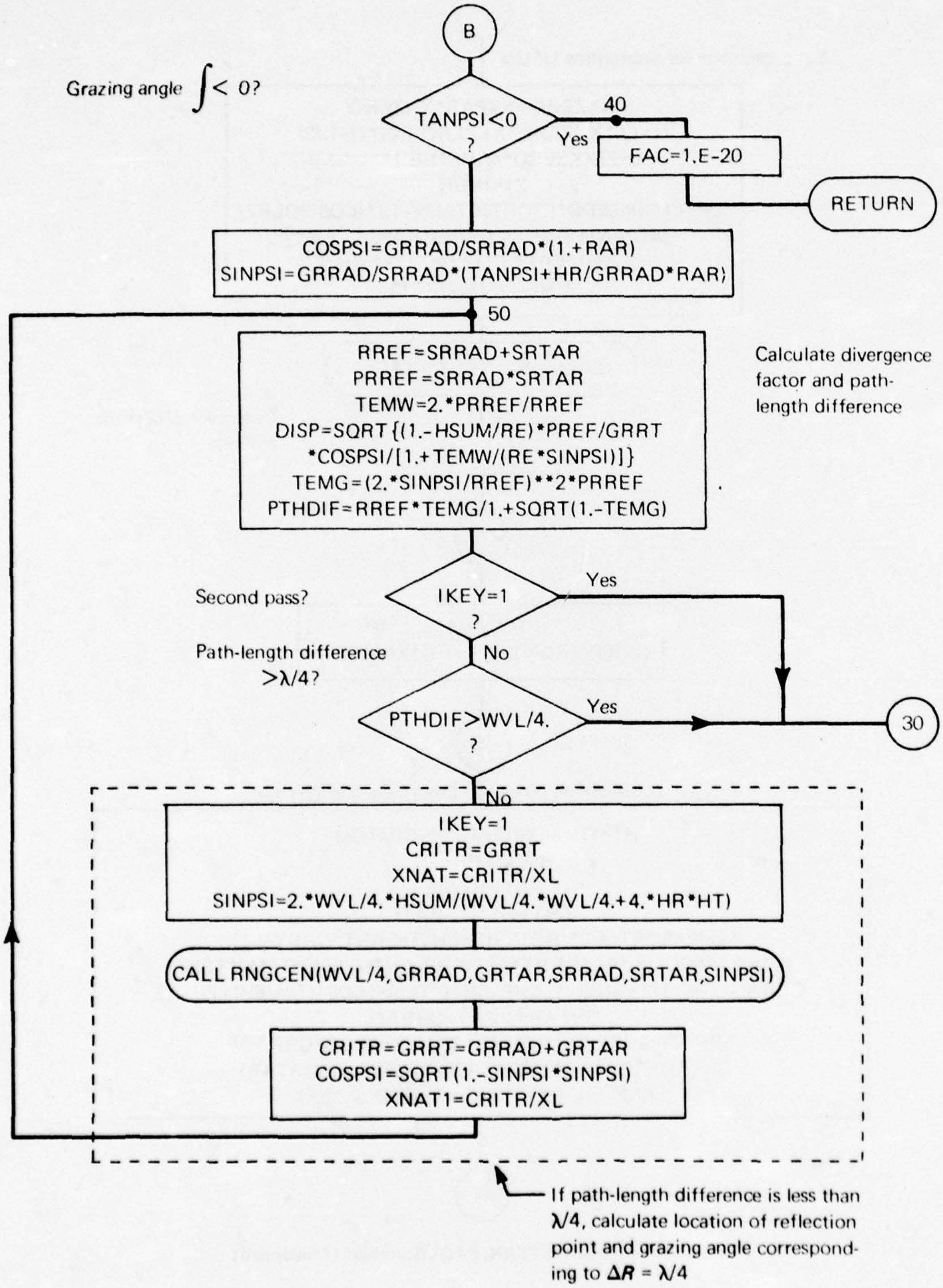


Fig. 3.12-6—MULPTH(ITAR,FAC) flowchart (Continued)

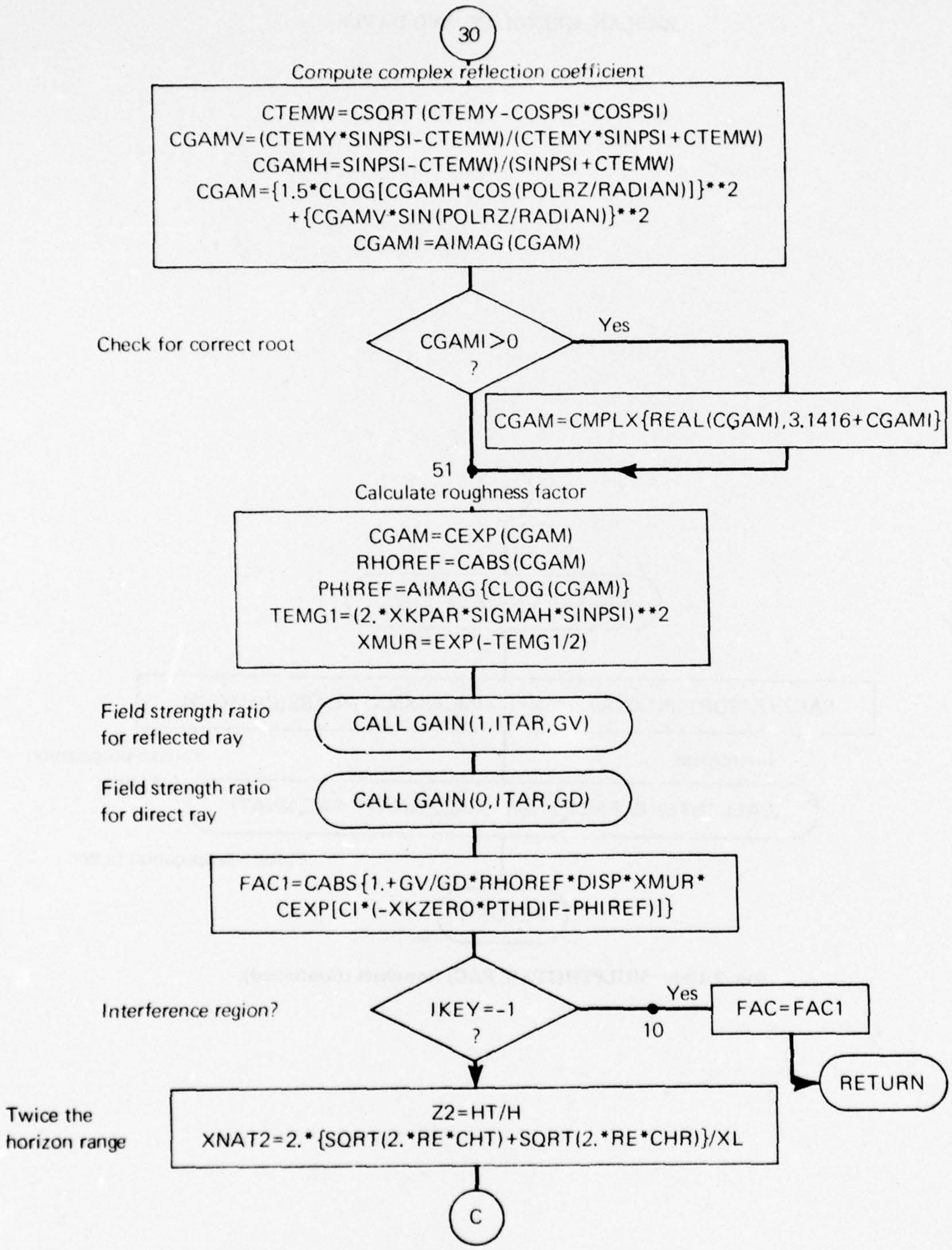


Fig. 3.12-6—MULPTH(ITAR, FAC) flowchart (Continued)

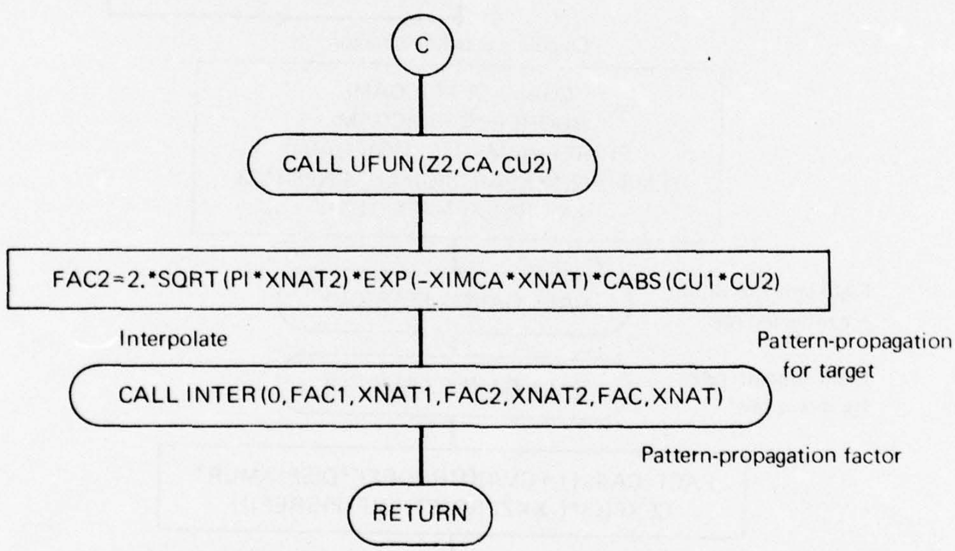


Fig. 3.12-6—MULPTH(ITAR, FAC) flowchart (Continued)

3.13 Subroutine NEWPOST

Subroutine NEWPOST is called by Subroutine NEXTSCN and the SURSEM Executive Routine. When called by NEXTSCN, it provides position information for the primary target (i.e., the target that is being considered for detection purposes); when called by the Executive Routine, it provides position information for all targets.

NEWPOST first determines the ratio of elapsed time to total target time according to

$$\Delta T = \frac{t - t_0(J)}{t_f(j) - t_0(J)} \tag{3.13-1}$$

where t is current time, $t_0(J)$ is the initial or starting time for target J, and $t_f(J)$ is the final time for target J.

The target coordinates with respect to the designated origin are computed from ΔT as follows:

$$x(k, J) = x_0(k, J) + \Delta T[x_f(k, J) - x_0(k, J)] \quad (k = 1, 2, 3) . \tag{3.13-2}$$

These coordinates are transformed to a radar-centered coordinate system by the following equation:

$$y(k, J) = x(k, J) - x_{\text{radar}}(k, J) \quad (k = 1, 2, 3) . \tag{3.13-3}$$

The range is then given by

$$R^2 = \sum_{k=1}^{3} [y(k, J)]^2 . \tag{3.13-4}$$

The target elevation with respect to the horizon (α) is determined from approximations made to the geometry of Fig. 3.13-1. The distance D can be found from

$$D = \left[y(1, J)^2 + y(2, J)^2\right]^{1/2} . \tag{3.13-5}$$

It can also be represented by

$$D = (2a_0h_1)^{1/2} + (2a_0h_a)^{1/2} \tag{3.13-6}$$

where a_0 is 4/3 of the Earth's radius. Equation (3.13-6) can now be solved for h_1 to yield

$$h_1 = \left[\frac{D}{95.2} - (ha)^{1/2}\right]^2 \tag{3.13-7}$$

where h_1 is in nautical miles. If it is assumed that triangle ABT is a right triangle, which is a good approximation for targets at ranges of less than 100,000 feet, then

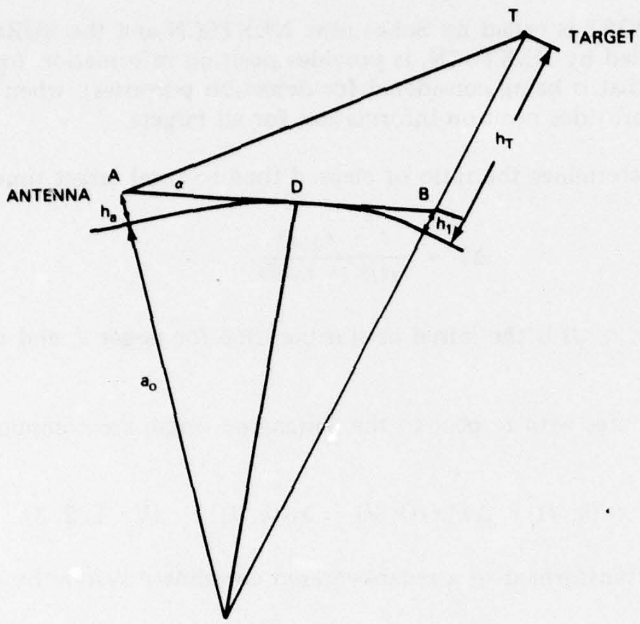


Fig. 3.13-1—Target geometry

$$\tan \alpha = \frac{h_T - h_1}{D} \tag{3.13-8}$$

or

$$\tan \alpha = \frac{h_T}{D} - \left(\frac{D^{1/2}}{95.2} - \frac{h_a^{1/2}}{D^{1/2}}\right)^2 . \tag{3.13-9}$$

Target azimuth θ is given by

$$\theta = \tan^{-1} \frac{y(2, J)}{y(1, J)} . \tag{3.13-10}$$

This completes the computational procedure in NEWPOST, and program control is returned to the calling subroutine.

Table 3.13-2 lists the variables used in Subroutine NEWPOST. The flowchart is shown in Fig. 3.13-3.

Table 3.13-2 — NEWPOST Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
BPRIME	tan α	Tangent of target elevation measured from the horizon
D	D	Ground range to target under consideration at time T (n.mi.)
DEL(I)		Coordinates of target J with respect to radar at time T (n.mi.):
	$y(1, J)$	$I = 1$, x coordinate
	$y(2, J)$	$I = 2$, y coordinate
	$y(3, J)$	$I = 3$, z coordinate
DOTP		Dot product of vectors defined by VEL(I) and DEL(I) or cosine of angle between the vectors
DSTAR		Distance from ship to horizon (n.mi.)
DT	ΔT	Ratio of time in flight to total flight time for target under consideration
DX	$y(1, J)$	x component of DEL(I) (n.mi.)
DY	$y(2, J)$	y component of DEL(I) (n.mi.)
DZ	$y(3, J)$	z component of DEL(I) (n.mi.)
NOLD		Number of targets considered on preceding pass
NT		Number of targets to be considered
SHIP(4)	$\sqrt{h_a}$	Square root of antenna height ($\sqrt{\text{n.mi.}}$)
T	t	Time for which the position of targets will be considered; set by NEXTSCN (s)
TOLD		Value of T on preceding pass (s)
TRGPOS(J,I)		Position of target J at time T (n.mi.)
	$x(1, J)$	$I = 1$, x coordinate
	$x(2, J)$	$I = 2$, y coordinate
	$x(3, J)$	$I = 3$, z coordinate
TRGPOS(J,4)	R	Slant range to target (n.mi.)
TRGPOS(J,5)	θ	Target azimuth (rad)
TRGPOS(J,6)	α	Target elevation measured from the horizon (rad)
TRGPOS(J,7)	D	Ground range to target J (n.mi.)
VEL(I)		Coordinates of target's final point with respect to its initial point (n.mi.):
		$I = 1$, x coordinate
		$I = 2$, y coordinate
		$I = 3$, z coordinate
VELMAG2		Magnitude of vector VEL(I)
XYZF(J,I)		Final position and time for target J
XYZI(J,I)		Initial position and time for target J (n.mi.):
	$x_0(1,J)$	$I = 1$, x coordinate
	$x_0(2,J)$	$I = 2$, y coordinate
	$x_0(3,J)$	$I = 3$, z coordinate
	$t_0(J)$	$I = 4$, time (s)

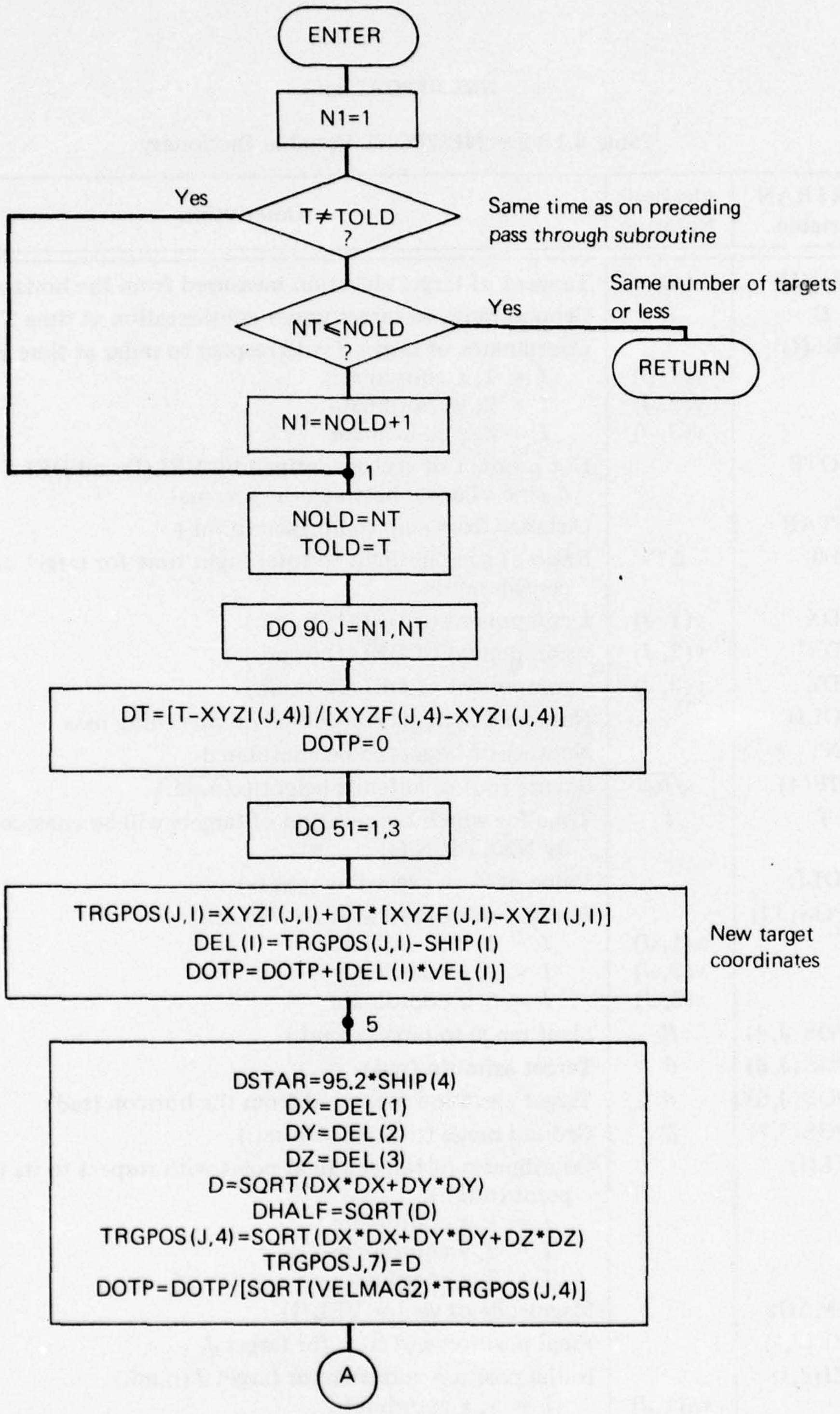


Fig. 3.13-3—NEWPOST(NT) flowchart

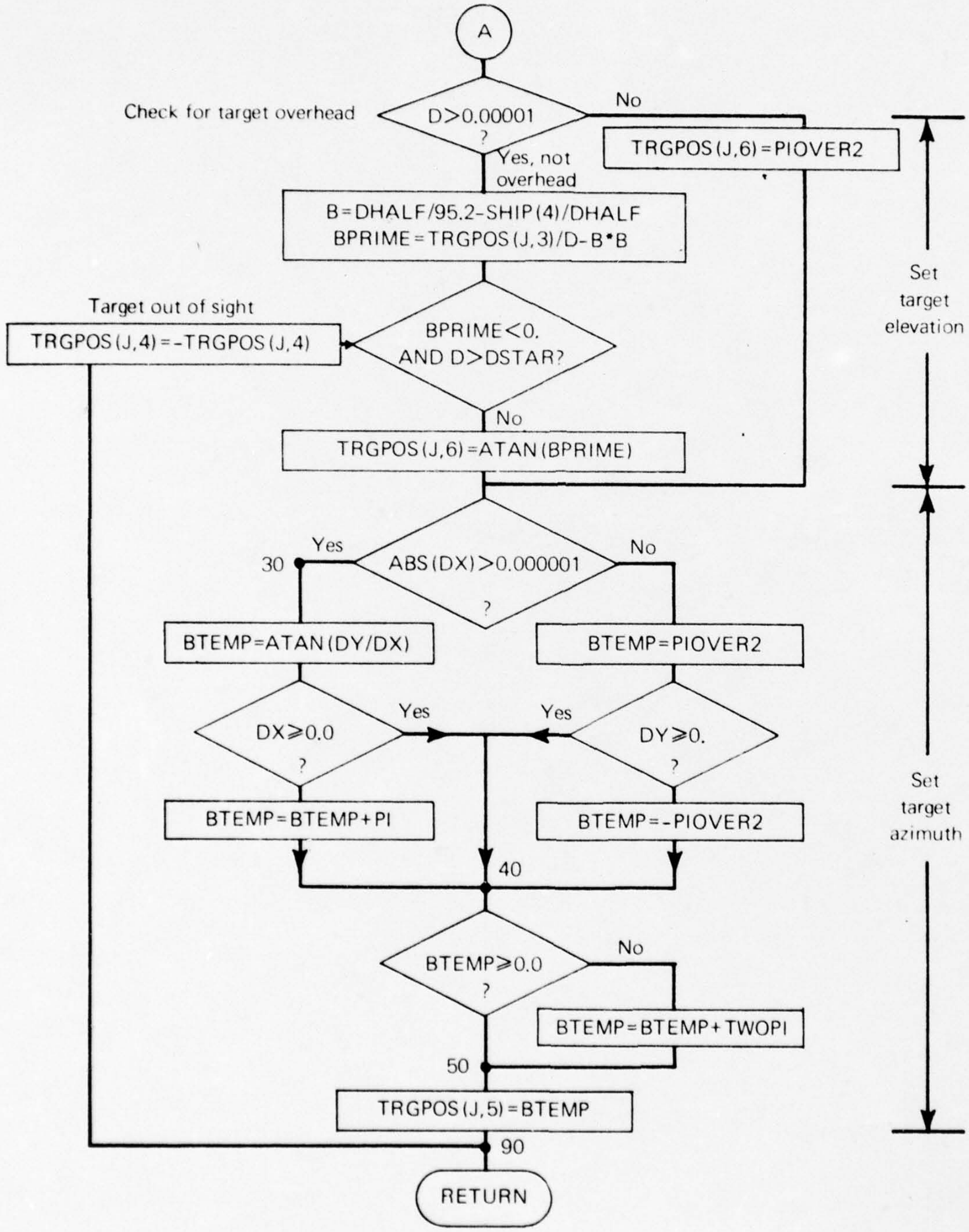


Fig. 3.13-3—NEWPOST(NT) flowchart (Continued)

3.14 Subroutine NEXTSCN

Subroutine NEXTSCN is called by the Executive Routine. Its primary function is to determine which scan mode will next try to see target 1 and to increment game time to correspond with this event. NEXTSCN also determines the range, azimuth, and elevation of target 1 at this instant.

This is accomplished by considering the time at which each scan mode will next see the target and by selecting the mode that has the shortest time until target illumination. Subroutine NEWPOST is then called to calculate the range, elevation, and azimuth of the target that correspond to the time at which the selected mode next sees the target. This essentially completes the process; however, checks must be made to ascertain that the target is (a) above the radar horizon, (b) within the 3-dB beamwidth for a pencil beam or within the first null for a cosecant-squared beam, (c) within instrumented range, and (d) beyond minimum range. If these conditions are satisfied, control of the program is returned to the Executive Routine. If not, the scan mode with the next shortest time until target illumination is addressed, and this procedure is continued until all conditions are met or the game time is exceeded.

See Table 3.14-1 for the variables used in NEXTSCN. The flowchart for this subroutine is shown in Fig. 3.13-2.

Table 3.14-1 — NEXTSCN Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
ENDTIME		Time at which game is ended (s)
NSCAN		Number of scan modes
RC(2)		Antenna pattern function indicator (0 = pencil beam, 1 = $\csc^2$ beam)
RC(5)		3-dB vertical beamwidth (rad)
RMODE(J,2)		Upper elevation coverage for mode J
RMODE(J,5)		Interlook period for mode J (s)
RMODE(J,7)		Instrumented range for mode J (n.mi.)
RMODE(J,9)		Next time at which scan mode J will see target (s)
SMODE(J,1)		Minimum range (n.mi.)
T		Current game time (s)
TMIN		Minimum value of RMODE(J,9) when all scan modes are considered (s)
TRGPOS(1,6)	E	Target elevation (rad)
TRGPOS(J,4)	R	Range to target J (n.mi.); set negative in NEWPOST if target is beyond horizon

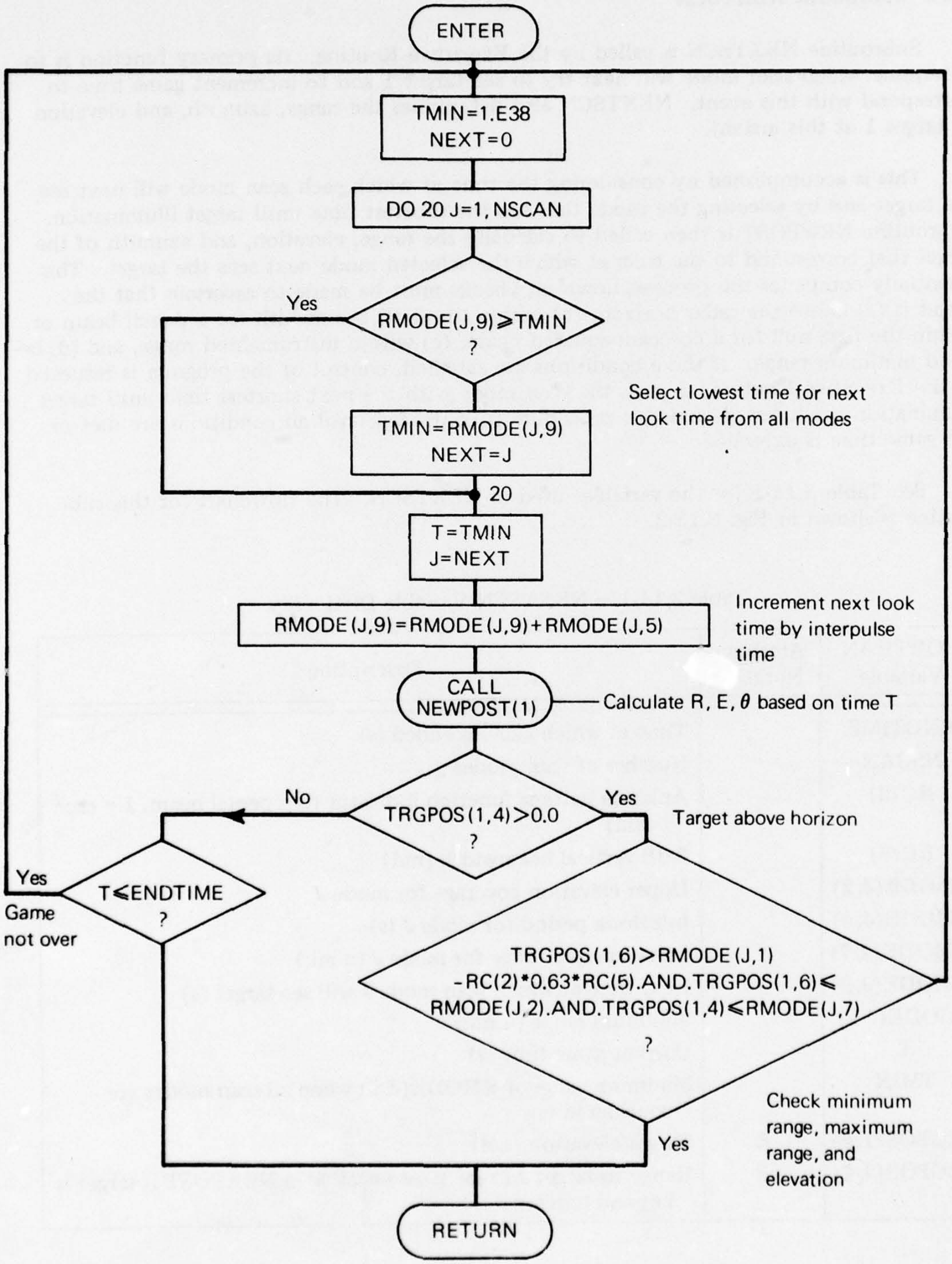


Fig. 3.14-2—NEXTSCN flowchart

3.15 Subroutine RNGCEN

Subroutine RNGCEN is called by Subroutine MULPTH. Its function is to determine the location of the reflection point on a 4/3 Earth's radius, corresponding to a path-length difference of $\lambda/4$. This information is utilized in evaluating the propagation factor in the intermediate region.

The subroutine is entered with a value of $\sin\psi$ that corresponds to the flat-Earth solution to the problem:

$$\sin\psi = 2\left(\frac{\lambda}{4}\right)\frac{h_r + h_t}{(\lambda/4)^2 + 4h_r h_t} \tag{3.15-1}$$

where $\lambda/4$ represents the path-length difference and h_r, h_t are defined in Fig. 3.15-1. This value of $\sin\psi$ is used to calculate the path-length difference Δ from the following equation, whose development will be discussed later:

$$\Delta = \frac{4\sin\psi}{\dfrac{1 + [1 + (2h_r/a\sin^2\psi)]^{1/2}}{h_r} + \dfrac{1 + [1 + (2h_t/a\sin^2\psi)]^{1/2}}{h_t}}\,. \tag{3.15-2}$$

The function f, defined as

$$f = \Delta - \frac{\lambda}{4}, \tag{3.15-3}$$

is evaluated, and its absolute value is compared to $\lambda/40$. If $|f| \geqslant \lambda/40$, the derivative of f with respect to $\sin\psi$ is determined and used to calculate a new value of $\sin\psi$:

$$\sin\psi_{\text{new}} = \sin\psi_{\text{old}} - f\left[\frac{1}{df/d(\sin\psi)}\right]. \tag{3.15-4}$$

A new value of Δ is then computed using $\sin\psi_{\text{new}}$, and this process is continued until the convergence criterion $|f| < \lambda/40$ is satisfied. When this occurs, the corresponding values of r_1, r_2, R_1, and R_2 are calculated and control of the program is returned to MULPTH.

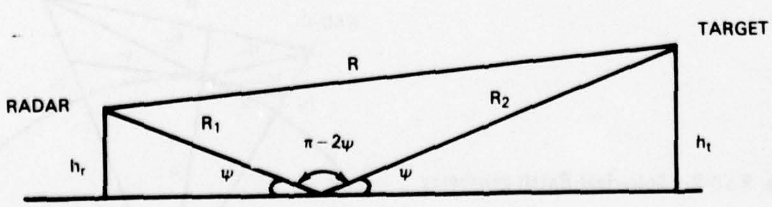


Fig. 3.15-1—Flat-Earth geometry

Equation (3.15-2) is developed as follows: From Fig. 3.15-2 it is seen that

$$R = \left[R_1^2 + R_2^2 + 2R_1R_2 \cos 2\psi\right]^{1/2} \tag{3.15-5}$$

$$= \left[(R_1 + R_2)^2 - 2R_1R_2(1 - \cos 2\psi)\right]^{1/2} \tag{3.15-6}$$

$$= \left[(R_1 + R_2)^2 - 4R_1R_2 \sin^2 \psi\right]^{1/2} . \tag{3.15-7}$$

Equation (3.15-7) can be rewritten as

$$R = (R_1 + R_2)\left[1 - \frac{4R_1R_2 \sin^2 \psi}{(R_1 + R_2)^2}\right]^{1/2}, \tag{3.15-8}$$

and since $\Delta = R - (R_1 + R_2)$,

$$\Delta = (R_1 + R_2)\left\{-1 + \left[1 - \frac{4R_1R_2 \sin^2 \psi}{(R_1 + R_2)^2}\right]^{1/2}\right\}, \tag{3.15-9}$$

or

$$\Delta = (R_1 + R_2)\,\frac{(4R_1R_2 \sin^2 \psi)/(R_1 + R_2)^2}{\left\{1 - \left[(4R_1R_2 \sin^2 \psi)/(R_1 + R_2)^2\right]\right\}^{1/2} + 1} . \tag{3.15-10}$$

When the path-length difference is close to $\lambda/4$, $\sin \psi \approx 0$, and consequently Eq. (3.15-10) can be simplified to

$$\Delta = \frac{2R_1R_2}{R_1 + R_2} \sin^2 \psi = \frac{2}{R_1^{-1} + R_2^{-1}} \sin^2 \psi . \tag{3.15-11}$$

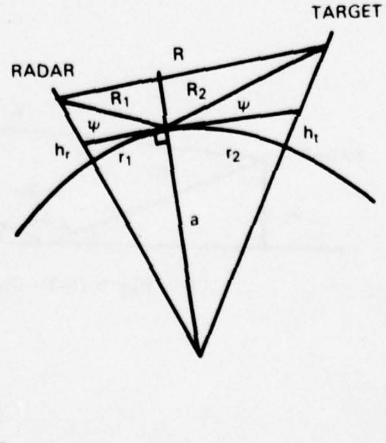


Fig. 3.15-2—Spherical-Earth geometry

In order to reduce Eq. (3.15-11) to Eq. (3.15-2), R_1 and R_2 must be expressed in terms of h_r, h_t, and $\sin\psi$. This is accomplished as follows: From Fig. 3.15-2 observe that

$$\sin\psi = \frac{h_r^2 - R_1^2 + 2ah_r}{2R_1 a} = \frac{h_r}{R_1}\left(\frac{h_r}{2a} + 1\right) - \frac{R_1}{2a}. \tag{3.15-12}$$

Neglecting $h_r/2a$ gives

$$\sin\psi = \frac{h_r}{R_1} - \frac{R_1}{2a}. \tag{3.15-13}$$

Applying the quadratic formula to Eq. (3.15-13) yields

$$R_1 = \frac{2h_r}{\sin\psi + [\sin^2\psi + (2h_r/a)]^{1/2}}. \tag{3.15-14}$$

Similarly

$$R_2 = \frac{2h_t}{\sin\psi + [\sin^2\psi + (2h_t/a)]^{1/2}}. \tag{3.15-15}$$

The above substitutions reduce Eq. (3.15-11) to Eq. (3.15-2).

The variables used in the RNGCEN subroutine are listed in Table 3.15-3. The flowchart is shown in Fig. 3.15-4.

Table 3.15-3 — RNGCEN Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
DELT	$\lambda/4$	Wavelength divided by four (m)
HR	h_r	Height of antenna (m)
HT	h_t	Height of target (m)
R1	R_1	Slant range from radar to reflection point (m)
R2	R_2	Slant range from reflection point to target (m)
RE	a	4/3 of the Earth's radius (m)
S	$\sin\psi$	Sine of the grazing angle
SR1	r_1	Ground range from radar to reflection point (m)
SR2	r_2	Ground range from reflection point to target (m)

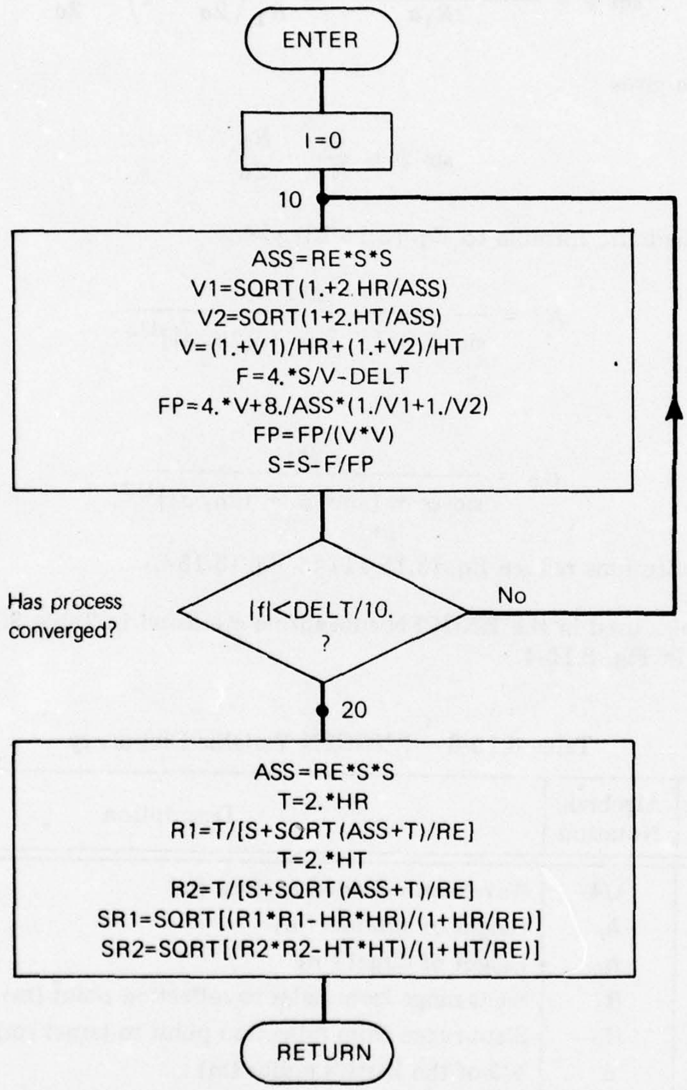


Fig. 3.15-4—RNGCEN(DELT,SR1,SR2,R1,R2,S) flowchart

3.16 Subroutine TARGETS

Subroutine TARGETS is called by the SURSEM Executive Routine. Its primary function is to read and list position and timing data for each target. The data are also modified for compatibility with other subroutines; that is, position data in thousands of feet are converted to nautical miles, time in seconds is converted to hours, and watts per megahertz are converted to watts per hertz.

Subroutine TARGETS also sets up a dummy target used to locate a clutter cell. This target is assigned the x and y coordinates, and the initial and final times of target 1. Its elevation is assumed to be zero.

In addition, the subroutine determines coefficients used in Eq. (3.16-1) for calculating the target's radar cross section as a function of aspect angle. This calculation is performed by Subroutine TARSIG. Equation (3.16-1) was developed under the assumption that the target's radar cross section, as viewed circumferentially, generates a lobing structure and that this structure can be represented by a linear combination of the functions cos 2θ, cos 4θ, and cos 8θ, where θ is the angle between the line of sight to the target and the target's broadside axis (see Fig. 3.17-3):

$$\sigma(\theta) = A_1 \cos 2\theta + A_2 \cos 4\theta + A_3 \cos 8\theta + A_4 . \qquad (3.16\text{-}1)$$

Expressions for the coefficients A_i as functions of the broadside, head-on, and minimum cross sections are developed as follows: First, the expression for $\sigma(\theta)$ is differentiated and expressed in terms of cos θ and sin 2θ, that is,

$$\sigma'(\theta) = 2A_1 \sin 2\theta + 8A_2(2\cos^2\theta - 1)\sin 2\theta$$
$$+32A_3(16\cos^6\theta - 24\cos^4\theta + 10\cos^2\theta - 1)\sin 2\theta . \qquad (3.16\text{-}2)$$

It is now assumed that the minimum radar cross section occurs at $\theta = \pi/3$, or 60 degrees off broadside. (This is consistent with measurements made on the type of aircraft that is normally encountered in applications of the program.) Setting $\sigma' = 0$ and $\cos\theta = -1/2$ in Eq. (3.16-2) yields

$$4A_3 - 2A_2 + A_1 = 0 . \qquad (3.16\text{-}3)$$

For $\theta = 0$, the radar cross section is given by the input value for the broadside cross section B. These values together with Eq. (3.16-3) reduce Eq. (3.16-1) to

$$\sigma(0) = 3A_2 - 3A_3 + 4A_2 = B . \qquad (3.16\text{-}4)$$

For $\theta = \pi/2$, the radar cross section is given by the input value for the normal cross section N, and hence

$$-A_2 + 5A_3 + A_4 = N . \qquad (3.16\text{-}5)$$

For $\theta = \pi/3$, the radar cross section is given by the input minimum value M. This yields

$$-3A_2 + 3A_3 + 2A_4 = 2M . \qquad (3.16\text{-}6)$$

Solving Eqs. (3.16-3) through (3.16-6) for A_1, A_2, A_3, and A_4 gives the following results:

$$A_4 = \frac{B + 2M}{2} , \qquad (3.16\text{-}7)$$

$$A_3 = \frac{B + 3N - 4A_4}{12} , \qquad (3.16\text{-}8)$$

$$A_2 = 5A_3 - N + A_4 , \qquad (3.16\text{-}9)$$

$$A_1 = 2A_2 - 4A_3 . \qquad (3.16\text{-}10)$$

Equations (3.16-7) through (3.16-10) are the expressions that are found at the end of the subroutine. The values generated are passed to Subroutine TARSIG for computing the radar cross section.

The reader is referred to Table 3.16-1 for a dictionary of the variables used in TARGETS. The flowchart for this subroutine is shown in Fig. 3.16-2.

Table 3.16-1 — TARGETS Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
A(I)	A_i	Weighting factors used in determining the radar cross section (RCS)
MODEL(J)		Marcum-Swerling cross-section model indicator
NUMTGT		Number of targets
SIGJAM(J)		Jamming energy for target J (W/MHz or W/Hz)
SIGTAR(I,J)		Radar cross section for target I (m^2):
	N	J = 1, head-on RCS
	B	J = 2, broadside RCS
	M	J = 3, minimum RCS
XYZF(J,K)		Final coordinates for target J:
		K = 1, x coordinate (10^3 ft or n.mi.)
		K = 2, y coordinate (10^3 ft or n.mi.)
		K = 3, z coordinate (10^3 ft or n.mi.)
		K = 4, time (s or hr)
XYZI(J,K)		Initial coordinates for target J

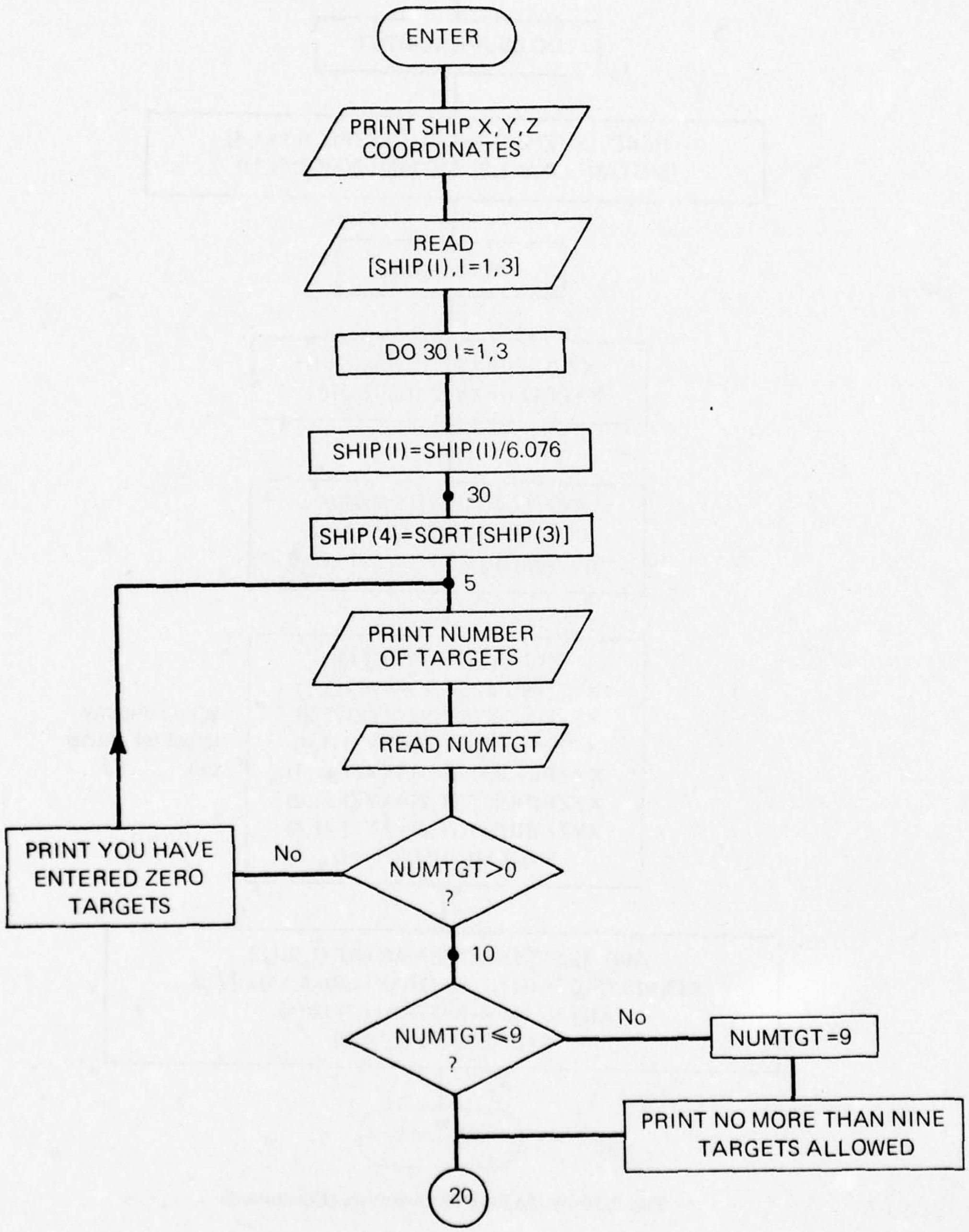


Fig. 3.16-2—TARGETS flowchart

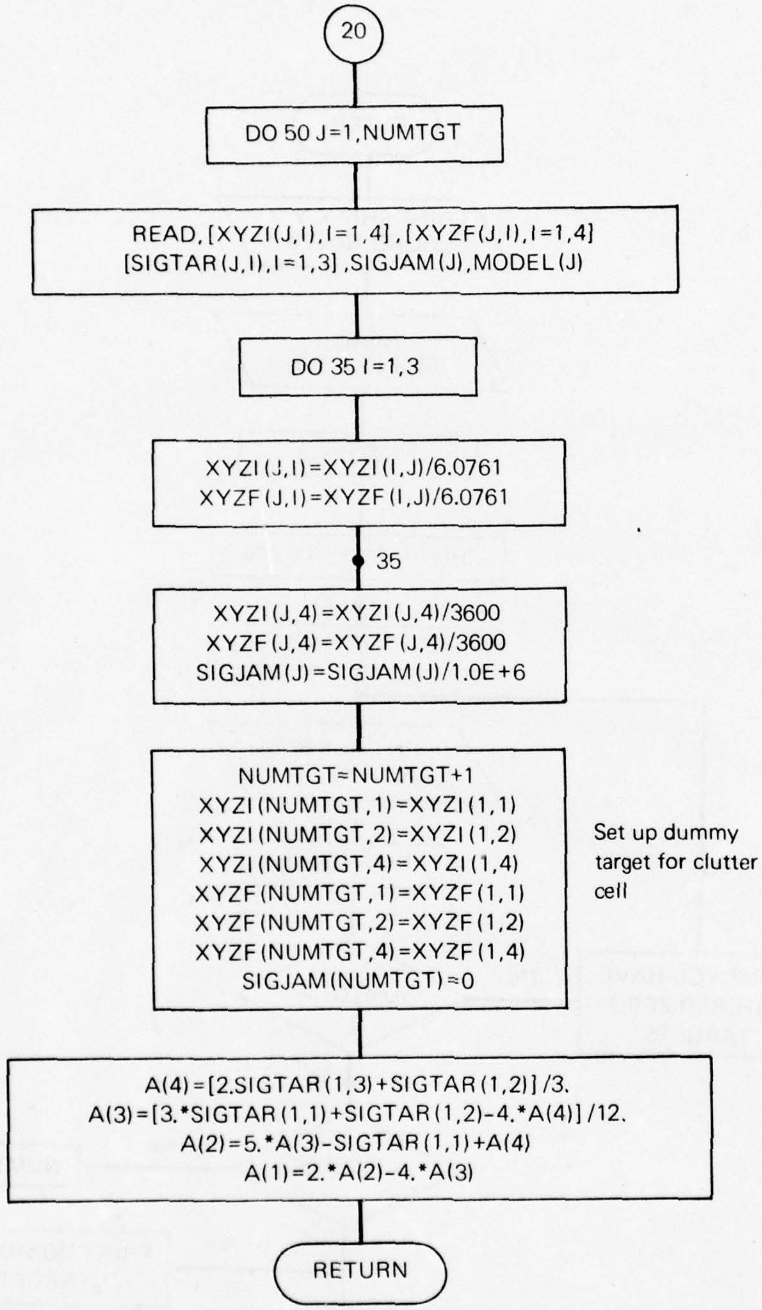


Fig. 3.16-2—TARGETS flowchart (Continued)

3.17 Subroutine TARSIG

Subroutine TARSIG (Table 3.17-1) is called by subroutine DETPROB. TARSIG's primary function is to determine the target's radar cross-section as a function of aspect angle. (See Fig. 3.17-2 for the subroutine flowchart.)

The radar cross section for an aspect angle θ is given by

$$F(\theta) = A_1 \cos 2\theta + A_2 \cos 4\theta + A_3 \cos 8\theta + A_4 .$$

The cos $2k\theta$ functions generate a lobing structure that is weighted by the A_i coefficients. The coefficients A_i are functions of the input values of the target's head-on and broadside radar cross-section, and the minimum radar cross-section that is encountered over the range of aspect angles. The angle θ is the angle between the line of sight to the target and the target's broadside axis (see Fig. 3.17-3), and the coefficients A_i are determined in subroutine TARGETS.

Table 3.17-1 — TARSIG Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
A(I)		Constants determined in Subroutine TARGETS
DOTP	$\cos \theta$	Cosine of the angle between the line of sight to the target and the target's velocity vector
TARCS	$\sigma(\theta)$	Target cross section (m^2)
TEMP(2)	$\cos 2\theta$	
TEMP(3)	$\cos 4\theta$	
TEMP(4)	$\cos 8\theta$	

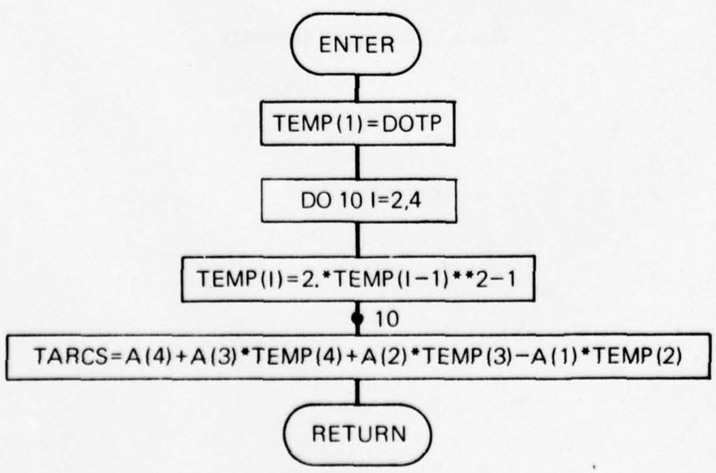


Fig. 3.17-2—TARSIG flowchart

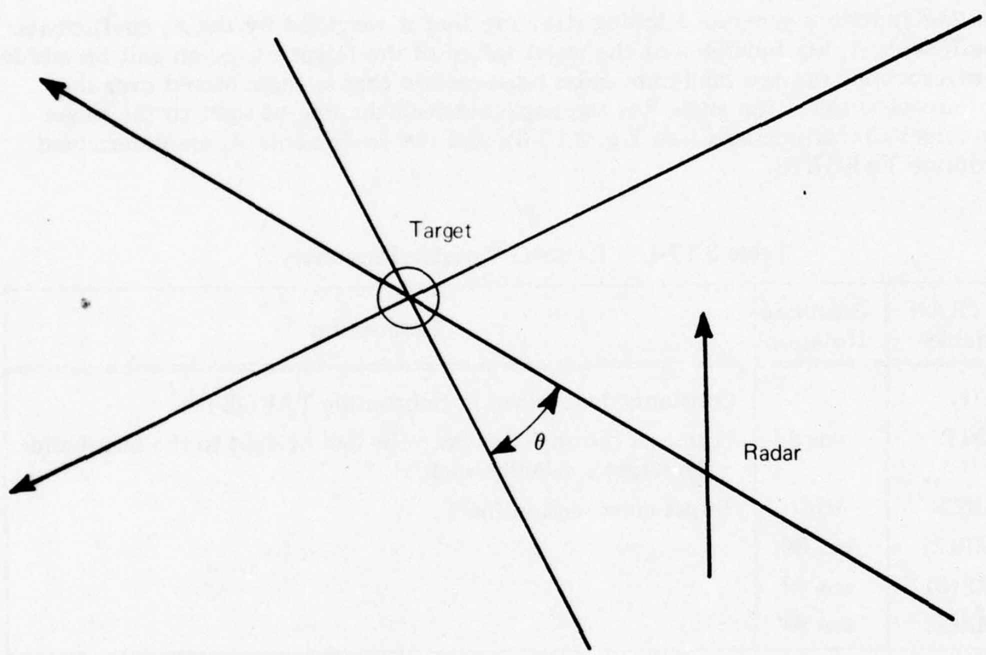


Fig. 3.17-3—Target geometry

3.18 Subroutine UFUN

Subroutine UFUN is called by Subroutine MULPTH to calculate the U functions that are used in determining the pattern-propagation factor F in the diffraction region. The equation as given by Kerr [8] for the pattern-propagation factor in the diffraction region is

$$F = 2(\pi X)^{1/2} e^{-C_1 X} |U_1(Z_1) U_1(Z_2)| ,$$

and UFUN evaluates the function $U_1(Z)$ for the values of Z given in its calling sequence. The term Z is alternatively assigned the height of the radar (Z_1) and the height of the target (Z_2) in the so-called natural units discussed in Section 3.12.

The rationale followed in evaluating $U(Z)$ is as follows: The term $U_1(Z)$ can be expressed in terms of the function $h_2(x)$. This is found (see Ref. 8, page 109) to be

$$U_1(Z) = i \frac{h_2(Z + A_1)}{h_2'(A_1)} .$$

The quantity A_1 is calculated in MULPTH and corresponds to CA in the calling sequence, and h_2' indicates the derivative of h_2.

The function $h_2(x)$ can in turn be expressed in terms of Airy functions:

$$h_2(x) = i 2^{4/3} 3^{1/6} Ai(x e^{\pi i/3}) .$$

Expressions for evaluating the Airy functions are found in Ref. 9. The Airy function is expressed as

$$Ai(z) = c_1 f(z) - c_2 g(z)$$

where $f(z)$ and $g(z)$ are given by power series for small values of $|z|$:

$$f(z) = 1 + \frac{1}{3!} z^3 + \frac{1 \cdot 4}{6!} z^6 + \frac{1 \cdot 4 \cdot 7}{9!} z^9 + \cdots$$

$$g(z) = z + \frac{2}{4!} z^4 + \frac{2 \cdot 5}{7!} z^7 + \frac{2 \cdot 5 \cdot 8}{10!} z^{10} + \cdots$$

For values of $|z| > 3$, $Ai(z)$ is approximated by an asymptotic expansion:

$$Ai(z) \approx \frac{\pi^{-1/2} z^{-1/4} e^{-\mu} \sum_{0}^{\infty} (-1)^k n_k \mu^{-k}}{2}$$

where

$$\mu = \frac{2z}{3}^{3/2}$$

$$n_0 = 1,$$

and

$$n_k = \frac{\Gamma(3k + 1/2)}{54^k k! \Gamma(k + 1/2)} .$$

The variables used in Subroutine UFUN are listed in Table 3.18-1. The calculation procedure is outlined in the macro flowchart shown in Fig. 3.18-2 and detailed in Fig. 3.18-3.

Table 3.18-1 — UFUN Variable Dictionary

FORTRAN Variable	Algebraic Notation	Description
CA	A_i	Function of complex dielectric constant and polarization, evaluated in MULPTH
CA1		$A_i e^{i\pi/3}$ (argument of Airy function)
CA13		$(A_1 e^{i\pi/3})^3$
CA14		$(A_1 e^{i\pi/3})^{-1/4}$
CAIR		The kth term of asymptotic expansion of Airy function with argument $(Z_i + A_1)e^{i\pi/3}$
CAIRP		The kth term of asymptotic expansion of derivative of Airy function with argument $(A_1)e^{i\pi/3}$
CANS	U_i	U function used to determine propagation factor in diffraction region
CC		$1/2\pi^{-1/2}(\mathrm{CZ})^{-1/4}e^{-\mu}$
CCP		$-1/2\pi^{-1/2}(\mathrm{CAI})^{1/4}e^{-\mu}$
CDA		$2/3(A_1 e^{i\pi/3})^{3/2}$
CDZ		$2/3[(Z_i + A_1)e^{i\pi/3}]^{3/2}$
CFI		$f[(Z_i + A_1)e^{i\pi/3}]$ summed to the kth term
CFIP		$f'[A_1 e^{i\pi/3}]$ summed to the kth term
CGI		$g[(Z_i + A_1)e^{i\pi/3}]$ summed to the kth term
CGIP		$g'[A_1 e^{i\pi/3}]$ summed to the kth term
CH2	$h_2(Z_1 + A_1)$	h Function used in determining U_1
CH2P	$h_2'(A_1)$	Derivative of h function
CI	i	$\sqrt{-1}$

Table 3.18-1 — UFUN Variable Dictionary (Continued)

FORTRAN Variable	Algebraic Notation	Description
COLDA		Previous value of CANS
CXP3		$e^{i\pi/3}$
CZ		$(Z_2 + A_1)e^{i\pi/3}$ (argument of Airy function)
CZ3		$[(Z_i + A_i)e^{i\pi/3}]^3$
CZ4		$[(Z_i + A_1)e^{i\pi/3}]^{-1/4}$
I	k	Counter indicating which term of the power series or asymptotic expansion is under consideration
IKEY		Indicator that determines which expansion to use for Airy functions: (0) $\|CZ\|$ and $\|CA1\| \leqslant 3$ (1) $\|CZ\| > 3$ (2) $\|CA1\| > 3$ (3) $\|CZ\|$ and $\|CA1\| > 3$
ISWIT		Indicator takes on values of 0, 1, where 0 = initial pass or new mode, 1 = same mode as previous pass
OLDZ		Value of Z on previous entry to UFUN
RTPI		$\sqrt{\pi}$
XC1	c_1	Constant used in evaluating Airy function
XC2	c_2	Constant used in evaluating Airy function
XMUL1		Coefficient of kth term of power series for $f(z)$
XMUL2		Coefficient of kth term of power series for $g(z)$
Z	Z_i	Height of point in space under consideration

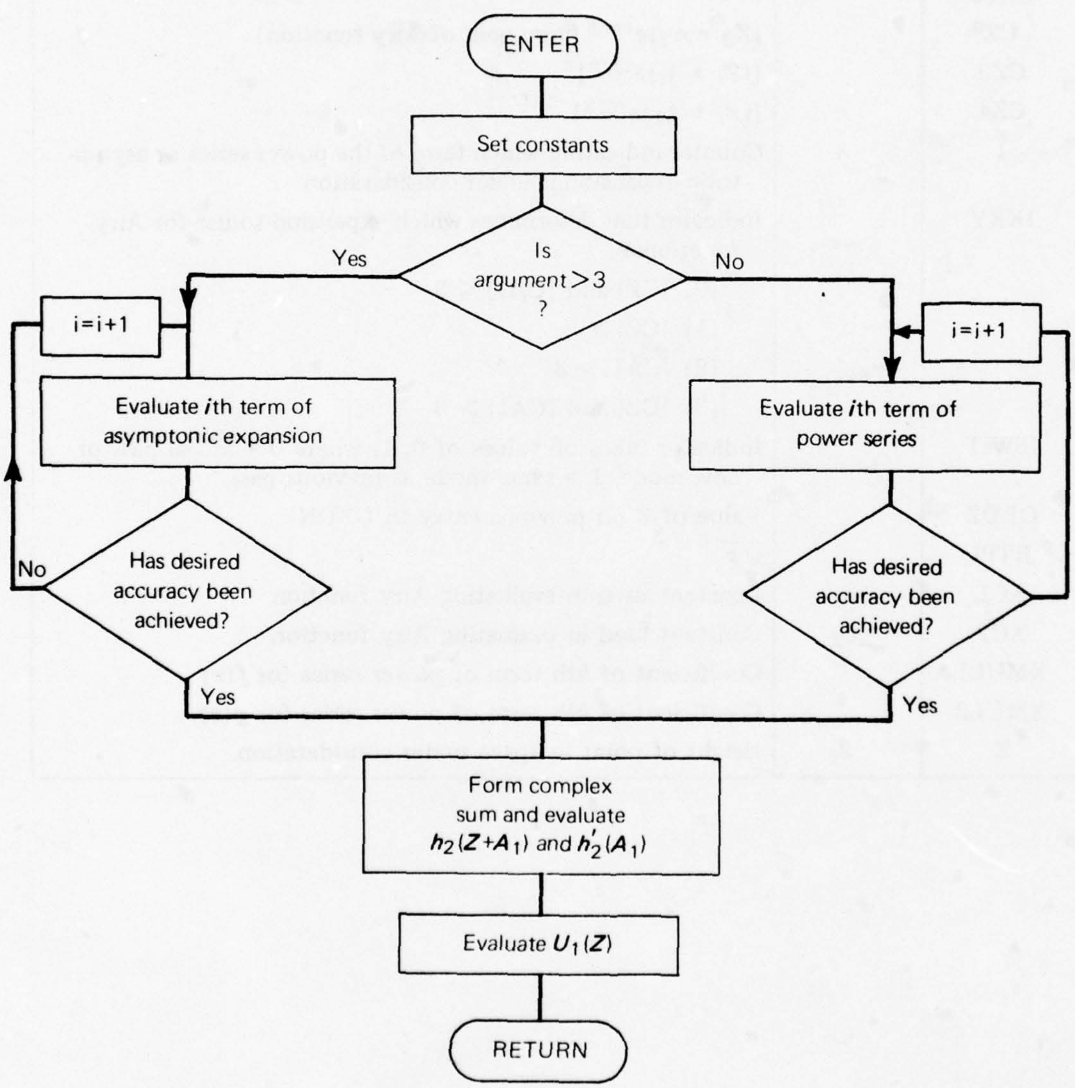


Fig. 3.18-2—UFUN macro flowchart

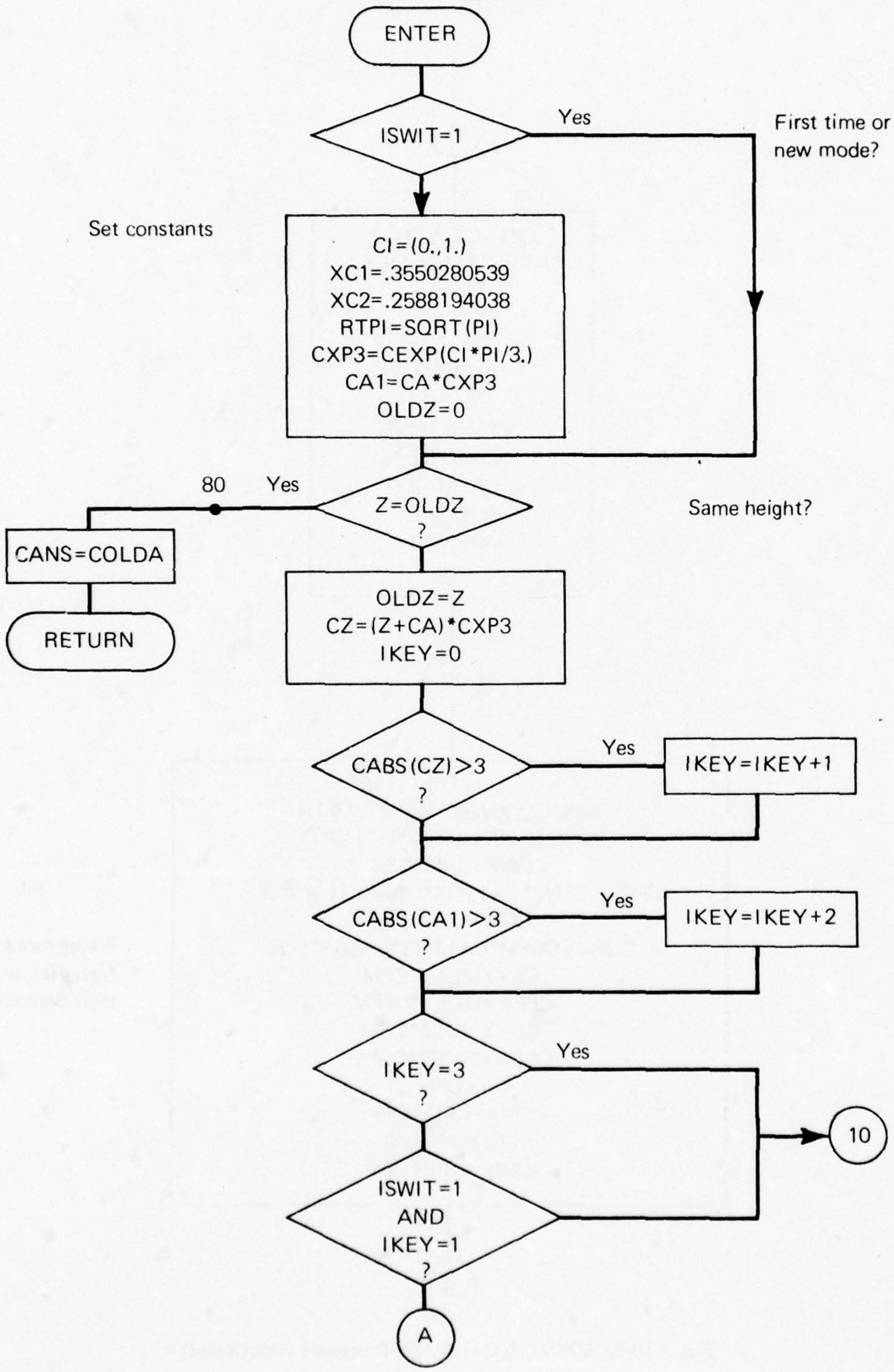


Fig. 3.18-3—UFUN(Z,CA,CANS) flowchart

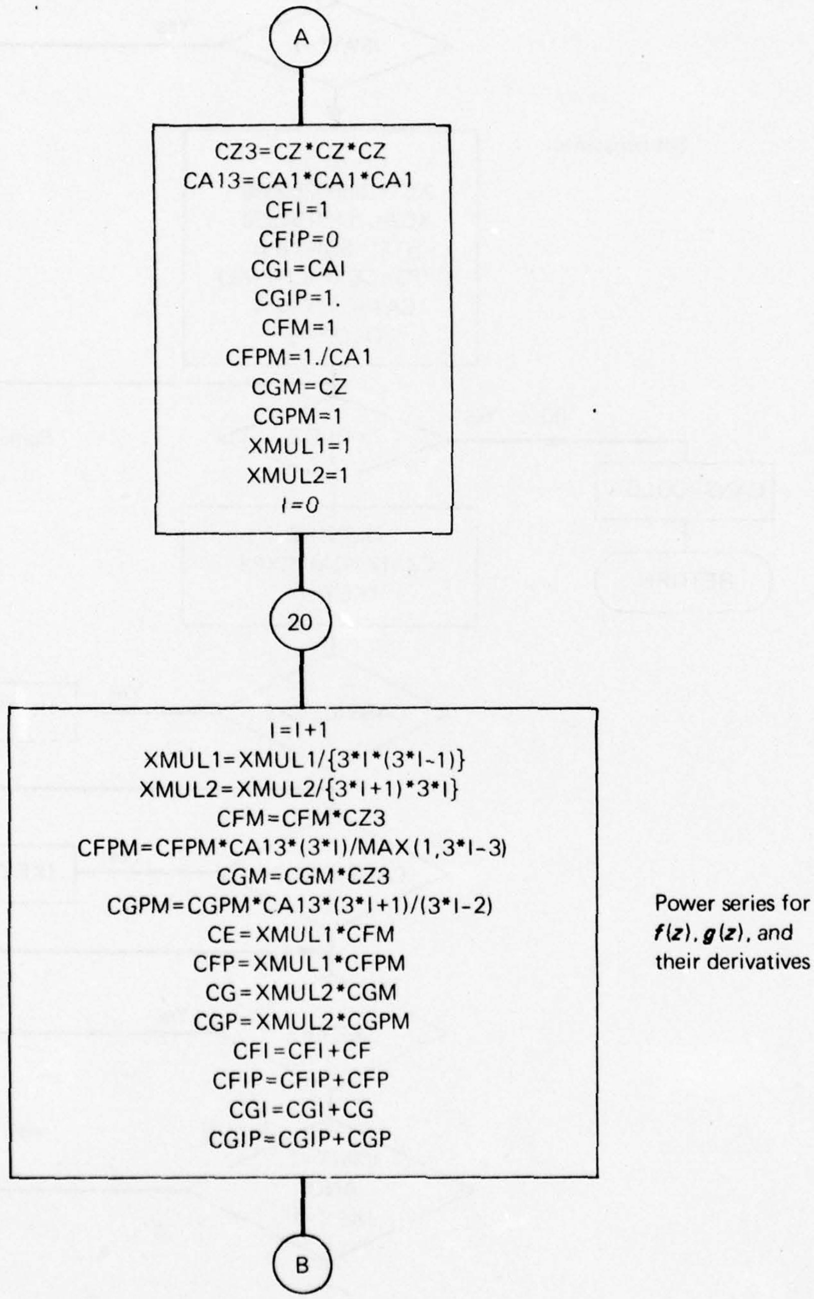


Fig. 3.18-3—UFUN(Z,CA,CANS) flowchart (Continued)

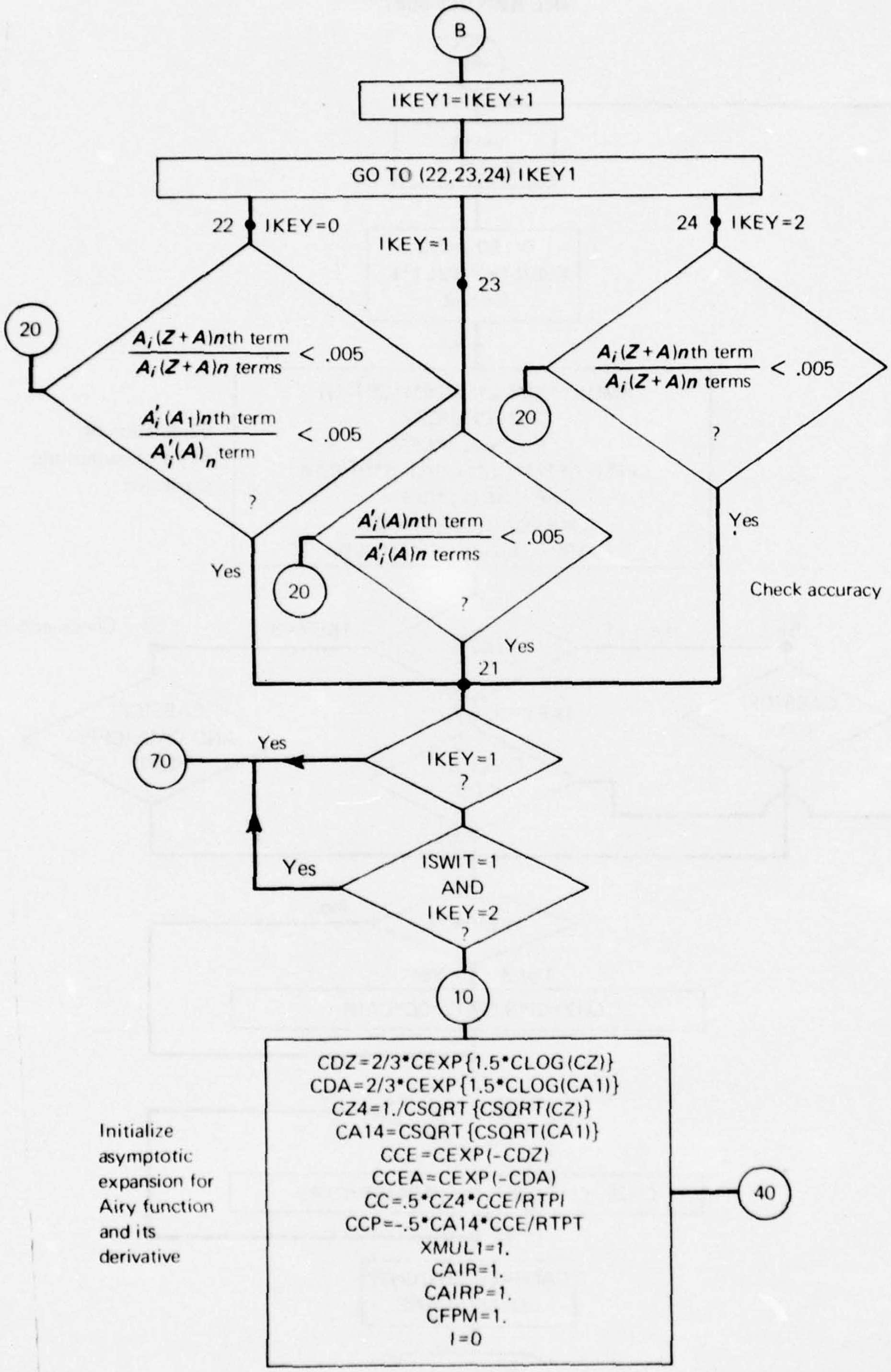


Fig. 3.18-3—UFUN(Z,CA,CANS) flowchart (Continued)

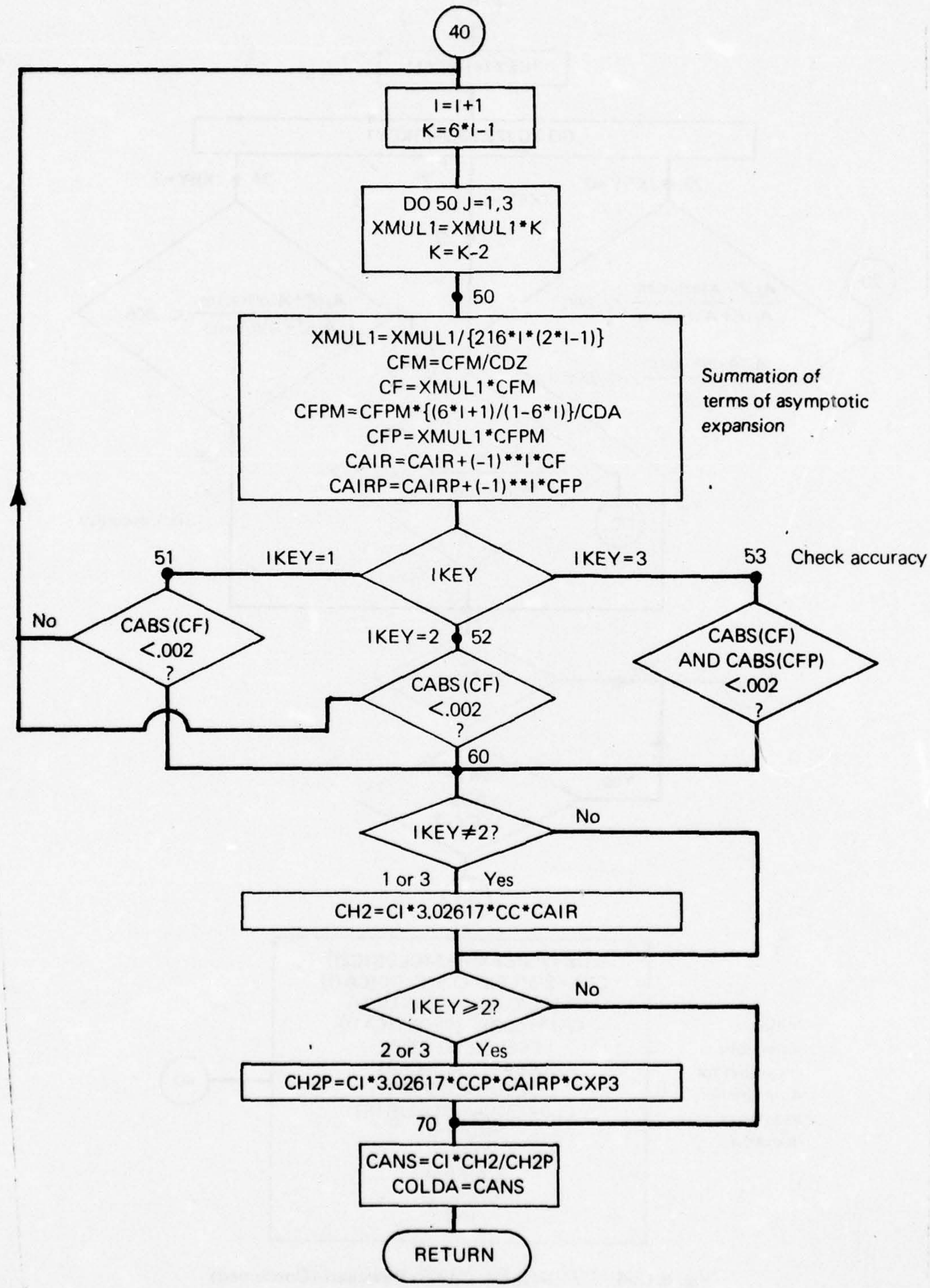


Fig. 3.18-3—UFUN(Z,CA,CANS) flowchart (Continued)

REFERENCES

1. J. I. Marcum and P. Swerling, "Studies of Target Detection by Pulsed Radar," *IRE Trans.* IT-6, No. 2 (1960).

2. W. J. Pierson, Jr., "Models of Random Seas Based on the Lagrangian Equations of Motion," New York University College of Engineering Technical Report prepared for ONR under Contract NONR-285(03), Apr. 1961.

3. M. I. Skolnik, *Introduction to Radar Systems*, McGraw-Hill Book Company, Inc., New York, 1962.

4. F. E. Nathanson, *Radar Design Principles*, McGraw-Hill, Book Company, Inc., New York, 1969.

5. R. W. Burling, "The Spectrum of Waves at Short Fetches," Dtsch. Hydrogr. Z., Vol. 12 (1959).

6. L. F. Fehlner, "Marcum's and Swerling's Data on Target Detection by a Pulsed Radar," Applied Physics Laboratory, Johns Hopkins University Report TG-451, Silver Spring, Md., July 1962.

7. L. V. Blake, "Machine Plotting of Radio/Radar Vertical-Plane Coverage," NRL Report 7098, June 25, 1970.

8. D. E. Kerr, editor, *Propagation of Short Radio Waves*, MIT Radiation Laboratory Series, Vol. 13, McGraw-Hill Book Company, Inc., New York, 1961.

9. M. Abramowitz and I. A. Stegun, editors, "Handbook of Mathematical Functions," National Bureau of Standards Applied Mathematics Series 55, U.S. Government Printing Office, Washington, D.C., June 1964.

Appendix A

SURSEM FORTRAN PROGRAM LISTING

```
C     THIS IS SURSEM, THE NRL SURVEILLANCE RADAR SYSTEM EVALUATION MODEL

C
C
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1       ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1       ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1        ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1       ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/D/ ALPHAD,SIGZ,V,AR1,AR2,SIGC,SIG
      COMMON/H/FAC4,AMBN,XJAMN,IKEYJG,XXXXX
      COMMON/I/ PBBS,HOFK,THETBK,DBDOWN,THH,THV,GN
      INTEGER ANS1,ANS2,ANS3,ANS4
      DIMENSION MCC(15)
      CONV=4.342944819
      RE = 8392375.04
      CNM = 161883.5474
      CCM = 300000000000.
      BETA = .718565021
      ACON = .607701593
      XKTOMS = .514444444
      XNMTOM = 1852.
      FOPISO = 157.9136706
      FOPIOB = 1984.401711
C     SPECIFY LOGICAL UNIT FOR OUTPUT SAVED DATA FILE(S)
      ITAPE = 1
   70 OFR=0
C
      TYPE 5
    5 FORMAT(1H0,'THIS IS SURSEM, THE NRL SURVEILLANCE RADAR SYSTEM EVA
     1LUATION MODEL.')
C
      TYPE 6
    6 FORMAT(/' DO YOU NEED A DESCRIPTION OF THE INPUT VARIABLES?   '$)
      ACCEPT 7, ANS3
    7 FORMAT(A3)
      IF(ANS3 .EQ. 3HYES) CALL INSTRC
C



C
      TYPE 9
    9 FORMAT(/' DO YOU WANT DETAILED OUTPUT?   '$)
      ACCEPT 7, ANS1
C
      CALL INITIAL
   10 CALL TARGETS
   11 CALL ENVIRN
      PS1=0.
      PS2=1.
      CUMP=1.
      DO 12 IC = 1,NSCAN
      MCC(IC)=IC
   12 CONTINUE
      MC=NSCAN
C
      IF(ANS1 .NE. 3HYES) GO TO 17
      TYPE 15
   15 FORMAT(1H0,'TYPE OUTPUT CONTROL PARAMETERS:   '$)
      ACCEPT*,PS1,PS2,CUMP,IREP,MC,(MCC(IC),IC=1,MC)
```

```
      IF(MC .NE. 0) GO TO 17
      MC=NSCAN
      MCC(1)=1
   17 CONTINUE
      IKEYF=0
      CALL MATCH
      ISTEP=2
      IREPF = 0
      PDCUM=0.0
      IF(ANS1 .NE. 3HYES) GO TO 20
      TYPE 40
   40 FORMAT(//' INDEX'6X'TIME'7X'BV'6X'SIGMA'5X'ESIG'4X'NCLT'5X'E/N'5X,
     1        'PDCUM'/6X'MODE'6X'RANGE'7X'BH'7X'FCTR'4X'NAMB'4X'NJAM'6X,
     2        'PDSS'/)
C
   20 CALL NEXTSCN
      IF(RC(1)+DWL(NEXT).NE.OFR)ISWIT=0
      OFR=RC(1)+DWL(NEXT)
      RFR=RC(1)/OFR
      RC(6)=GN/RFR**2
      RC(4)=THH*RFR
      RC(5)=THV*RFR
      PBBS = RMODE(NEXT,1) + RC(5)/2.
      TSEC=T*3600.0
      IF(T.GT.ENDTIME) GO TO 110
      DO 21 IC=1,MC
      IF(NEXT.EQ.MCC(IC))GO TO 22
   21 CONTINUE
      GO TO 20
   22 CONTINUE
      CALL NEWPOST(NUMTGT)
      ALPHAD = 0
      SIGZ = 0
      SIGC = 0
      SIG = 0
      AR1 = 0
      AR2 = 0
      V = 0
      CALL DETPROB(PDSS)
      IF(ISTEP.EQ.2)CUM(1,2)=PDCUM
      PDCUM=1.0-(1.0-PDCUM)*(1.0-PDSS)
      IF(PDSS.GE..001)GO TO 42
      GO TO 43
   42 CONTINUE
      CUM(ISTEP,1)=TSEC
      CUM(ISTEP,2)=PDSS
      ISTEP = ISTEP+1
   43 CONTINUE
      IF(ANS1 .NE. 3HYES) GO TO 80
      IF(PDSS.LT.PS1) GO TO 80
      IF(IKEYF.EQ.0)GO TO 145
      IF(PDSS.GT.PS2) GO TO 80
      IF(PDCUM.GT.CUMP) GO TO 80
  145 CONTINUE
      IKEYF=1
      IST = ISTEP-1
      IREPF = IREPF+1
      IF(MOD(IREPF,IREP).NE.0) GO TO 80
      BVDEG=TRGPOS(1,6)*RADIAN
      BHDEG=TRGPOS(1,5)*RADIAN
      DBE=CONV*ALOG(RC(12))
      DBN=CONV*ALOG(RC(14))
```

```
      SN=DBE-DBN
      TRGP = TRGPOS(1,4)*6.0802
      AMBNDB = CONV*ALOG(AMAX1(AMBN,1.0E-35))
      XJAMNDB = CONV*ALOG(AMAX1(XJAMN,1.0E-35))
      SCDB = CONV*ALOG(AMAX1(SIGC,1.0E-35))
      SFAC4=CONV*ALOG(AMAX1(FAC4,1.0E-35))
C
      TYPE 50,IST,TSEC,BVDEG,TARCS,DBE,SCDB,SN,PDCUM,
     1          NEXT,TRGP,BHDEG,SFAC4,AMBNDB,XJAMNDB,PDSS
   50 FORMAT(I5,F11.1,F10.2,F10.1,F9.0,F8.0,F9.2,F9.3/I9,F12.1,F10.1,
     1       F10.0,2F8.0,F10.3/)
   80 CONTINUE
      IF (ISTEP.LE.4000) GO TO 20
      TYPE 90
   90 FORMAT(1H0,' PROBABILITY OF DETECTION TABLE EXCEEDED. RUN TERMINAT

     1ED.'/)
  110 ISTEP=ISTEP-1
      CUM(1,1) = ISTEP
      TYPE 115
  115 FORMAT(1H0,'DO YOU WANT TO SAVE DATA?   '$)
      ACCEPT 7,INP
      IF(INP .EQ. 2HNO) GO TO 301
C     WRITE FILE
      WRITE(ITAPE,118) ISTEP, (CUM(I,1),CUM(I,2),I=1,ISTEP)
  118 FORMAT(1X,I4/(1X,5(F7.1,F7.5)))
C
  301 CONTINUE
      XJAMN=0.
      TYPE 120
  120 FORMAT(/' IS THERE ANOTHER ENVIRONMENT?    '$)
      ACCEPT 7,IANS
      IF(IANS .EQ. 3HYES) GO TO 11
C
      TYPE 122
  122 FORMAT(/' IS THERE ANOTHER SHIP/TARGET SET?    '$)
      ACCEPT 7,ANS2
      IF (ANS2 .EQ. 3HYES) GO TO 10
C
      TYPE 125
  125 FORMAT(/' IS THERE ANOTHER SET OF RADAR PARAMETERS?    '$)
      ACCEPT 7,ANS2
      IF (ANS2 .EQ.3HYES) GO TO 70
  140 CONTINUE
C
      TYPE 150
  150 FORMAT(/' PROGRAM FINISHED.'/)
  130 CONTINUE
      STOP
      END


*************************************************************


      SUBROUTINE INSTRC
C
C     PRINT INPUT INFORMATION
C
      TYPE 10
   10 FORMAT(1H0,' THERE ARE 11 BASIC RADAR INPUT PARAMETERS'/'    1 RAD

     1AR FREQUENCY IN MHZ'/'    2 ANTENNA PATTERN FUNCTION INDICATOR'/'
```

```
     2       0 = PENCIL BEAM'/'        1 = COSECANT SQUARE BEAM'/'    3
     3RECEIVER NOISE IN DB'/'    4 HORIZONTAL 3DB BEAMWIDTH IN DEGREES'/
     4'   5   VERTICAL 3DB BEAMWIDTH IN DEGREES'/'    6 ONE-WAY ANTENN
     5A GAIN IN DB'/'    7 ONE-WAY SIDELOBE LEVEL IN DB DOWN'/'    8 REC
     6EIVER LOSS IN DB'/'    9 TRANSMITTER LOSS IN DB'/'   10 NUMBER OF
     7SCAN MODES'/'   11 LINEAR POLARIZATION IN DEGREES'/'        0 = HO
     8RIZONTAL'/'       90 = VERTICAL')
C
      TYPE 15
   15 FORMAT(1H0,' FOR EACH RADAR SCAN MODE, THERE ARE 16 INPUT PARAMETE
     1RS'/'  1+2 LOWER AND UPPER BOUNDARY FOR ELEVATION ANGLE COVERAGE I
     2N DEGREES'/'    3 PEAK POWER IN MW'/'    4 PULSE LENGTH IN MICROSE
     3C'/'    5 INTERLOCK PERIOD IN SEC'/'    6 SCAN OFFSET IN SEC'/'
     4 7 INSTRUMENTED RANGE IN NMI'/'    8 MODE DEPENDENT LOSS IN DB'/'
     5   9 NUMBER OF PULSES INTEGRATED'/'   10 MINUS LOG10 (FALSE ALARM
     6PROBABILITY)'/'   11 COMPRESSED PULSE LENGTH IN MICROSEC'/'   12 S
     7EA CLUTTER IMPROVEMENT FACTOR IN DB'/'    13 I.F. BANDWIDTH IN MH.
     8 IF 0, BANDWIDTH WILL BE'/'            SET AT 1.0/(COMPRESSED PULSE LE
     9NGTH)'/'    14 MODE DEPENDENT FREQUENCY INCREMENT IN MHZ'/'   15 BL
     1ANKING TIME IN MICROSECONDS.  IF 0,'/'           SET AT PULSE LENGTH'
     2/'   16 RAIN CLUTTER IMPROVEMENT FACTOR IN DB')
C
      TYPE 20
   20 FORMAT(1H0,' THERE ARE 13 INPUT PARAMETERS FOR EACH TARGET'/'  1-4
     1 INITIAL COORDINATES (X,Y,Z,T) INKFT AND SEC'/'  5-8 TERMINAL COO
     2RDINATES (,Y,Z,T) IN KFT AND SEC'/' 9-11 RADAR REFLECTIVE AREAS F
     3OR HEAD-ON,'/'          BROADSIDE, AND MINUMUM IN SQ. METERS'/'  12
     4 JAMMING POWER DENSITY IN W/MHZ'/'   13 MARCUM SWERLING CROSS SECT
     5ION MODEL')
C
      TYPE 25
   25 FORMAT(1H0,' THERE ARE 4 ENVIRONMENTAL INPUT PARAMETERS'/'    1 WI
     1ND SPEED IN KNOTS'/'    2 HEIGHT OF WIND SPEED MEASUREMENT IN KFT'
     2/'    3 MULTIPATH INDICATOR'/'          1 = MULTIPATH'/'        0 =
     3NO MULTIPATH'/'    4 RAINFALL RATE IN MM/ HR')
C
      TYPE 30
   30 FORMAT(1H0,' THERE ARE SEVERAL PRINT CONTROL PARAMETERS'/'  1+2 LO
     1WER AND UPPER BOUNDS FOR WHICH SINGLE SCAN'/'         PROBABILITY O
     2F DETECTION WILL BE LISTED'/'    3 LARGEST CUMULATIVE PROBABILITY
     3OF DETECTION'/'          TO BE LISTED'/'    4 PRINT FREQUENCY INDICA
     4TOR.  IF THIS NUMBER WERE 5, THEN ONE OUT'/'        OF EVERY 5 LIN
     5ES OF POSSIBLE OUTPUT WOULD BE LISTED'/'    5 NUMBER OF MODES TO B
```

```
     6E USED'/'     6 MODE NUMBERS TO BE USED'/1H0)
C
      RETURN
      END


**************************************************************


      SUBROUTINE INITIAL
      DIMENSION AMODE(15,2)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1     DOTP,POLRZ,IKEYF,XKTCMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/I/ PBBS,HOFK,THETBK,DBDOWN,THH,THV,GN    .
      RC(10)=290.0*1.38*10.0**(-23)
      MILLION=1.0E+6
      PI=3.1415926536
      TWOPI=PI*2.0
      PIOVER2=PI/2.0
      RADIAN=57.29578
C
      TYPE 5
    5 FORMAT(1H0,'TYPE 11 BASIC RADAR INPUTS:    '$)
      ACCEPT*,(RC(I),I=1,9),NSCAN,POLRZ
C
      IF(NSCAN .LE. 15) GO TO 10
      TYPE 6
    6 FORMAT('    NO MORE THAN 15 SCAN MODES WILL BE ACCEPTED')
      NSCAN = 15
   10 CONTINUE
      RC(4)=RC(4)/RADIAN
      RC(5)=RC(5)/RADIAN
      THH=RC(4)
      THV=RC(5)
      RC(6)=10.**(RC(6)/10.)
      GN=RC(6)
      RC(3)=10.**(RC(3)/10.)
      RC(7)=10.**(-RC(7)/20.)
      RC(8)=10.**(-RC(8)/10.)
      RC(9)=10.**(-RC(9)/10.)
      DBDOWN = RC(7)*RC(7)
      DO 80 J=1,NSCAN
C
      TYPE 50,J
   50 FORMAT(/'   TYPE 16 INPUTS FOR MODE',I3,':   '$)
      ACCEPT*,(RMODE(J,I),I=1,8),AMODE(J,1),PMF,RMODE(J,11),SUBC(J),
     1        RMODE(J,12),DWL(J),SM,SMODE(J,2)
      IMODE(J,1)=AMODE(J,1)
      MPF=PMF
      IF (RMODE(J,5).GT.0.0) GO TO 55
      RMODE(J,5)=10.0
   55 RMODE(J,1)=RMODE(J,1)/RADIAN
      RMODE(J,2)=RMODE(J,2)/RADIAN
      IF(SM.EQ.0)SM=RMODE(J,4)
      SMODE(J,1)=153.*SM/XNMTOM
      SMODE(J,2) = 10.**(-SMODE(J,2)/10.)
      SMODE(J,3) = PMF
      RMODE(J,5)=RMODE(J,5)/3600.0
```

```
      RMODE(J,6)=RMODE(J,6)/3600.0
      RMODE(J,4)=RMODE(J,4)/MILLION
      RMODE(J,8)=10.**(-RMODE(J,8)/10.)
      RMODE(J,3)=RMODE(J,3)*MILLION
      TAU(J) = RMODE(J,11)/MILLION
      IF(RMODE(J,12).LE.0) RMODE(J,12) = 1.0/RMODE(J,11)
      RMODE(J,12) = RMODE(J,12)*MILLION
      IMODE(J,2)=MAX1(RMODE(J,12)*RMODE(J,11)/MILLION+0.5,1.0)
      RMODE(J,11) = 10.**(-PMF)
      SUBC(J) = 10.**(-SUBC(J)/10.)
   80 CONTINUE
      RETURN
      END


**************************************************************


      SUBROUTINE TARGETS
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1      ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1    . ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1      ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1      DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/E/ A(4)
      COMMON/DET/MODEL(10)
C
C     X Y Z SHIP COORDINATES
      TYPE 2
    2 FORMAT(1H0,'TYPE SHIP/RADAR POSITION COORDINATES (X,Y,Z)',
     1' IN KFT:   '$)
      ACCEPT*,(SHIP(I),I=1,3)
      DO 3 I = 1,3
         SHIP(I) = SHIP(I)/6.0802
    3 CONTINUE
      SHIP(4)=SQRT(SHIP(3))
C     TARGETS
C
    4 TYPE 5
    5 FORMAT(1H0,'TYPE NUMBER OF TARGETS:   '$)
      ACCEPT*,NUMTGT
C
      IF(NUMTGT .GT. 0) GO TO 15
      TYPE 10
   10 FORMAT('   YOU HAVE ENTERED ZERO TARGETS')
      GO TO 4
C
C     9 TARGETS MAXIMUM
   15 IF(NUMTGT .LE. 9) GO TO 25
      TYPE 20
   20 FORMAT(' NO MORE THAN 9 TARGETS WILL BE ACCEPTED')
      NUMTGT = 9
C
   25 DO 50 J = 1,NUMTGT
         TYPE 30, J
   30    FORMAT(/'  TYPE 13 PARAMETERS FOR TARGET ',I2,':   '$)
    .    ACCEPT*,(XYZI(J,I),I=1,4),(XYZF(J,I),I=1,4),(SIGTAR(J,I),I=1,3)

     1            ,SIGJAM(J),MODEL(J)
      DO 35 I=1,3
      XYZI(J,I) = XYZI(J,I)/6.0802
```

```
      XYZF(J,I) = XYZF(J,I)/6.0802
   35 CONTINUE
      XYZI(J,4)=XYZI(J,4)/3600.
      XYZF(J,4)=XYZF(J,4)/3600.
      SIGJAM(J) = SIGJAM(J)/1.0E+6
   50 CONTINUE
      NUMTGT =NUMTGT+1
      XYZI(NUMTGT,1) =XYZI(1,1)
      XYZI(NUMTGT,2) = XYZI(1,2)
      XYZI(NUMTGT,4) = XYZI(1,4)
      XYZF(NUMTGT,1) = XYZF(1,1)
      XYZF(NUMTGT,2) = XYZF(1,2)
      XYZF(NUMTGT,4) = XYZF(1,4)
      SIGJAM(NUMTGT) = 0
      A(4) = (2.*SIGTAR(1,3) + SIGTAR(1,2))/3.
      A(3) = (3.*SIGTAR(1,1) + SIGTAR(1,2) - 4.*A(4))/12.
      A(2) = 5.*A(3) - SIGTAR(1,1) + A(4)
      A(1) = 2.*A(2) -4.*A(3)
      RETURN
      END


*************************************************************


      SUBROUTINE ENVIRN
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPI03,FOPISO
      COMMON/H/FAC4,AMBN,XJAMN,IKEYJG,XXXXX
C
      TYPE 10
   10 FORMAT(1H0,'TYPE 4 ENVIRONMENTAL PARAMETERS:   '$)
      ACCEPT*,(ENVIR(I),I=1,4)
      FAC4=1.
      ENVIR(2) = ENVIR(2)/6.0802
      SIGMAH = .00666667*(XKTOMS*ENVIR(1))**2*(XNMTOM*ENVIR(2)/10.)
     1        **(-.19364)
      XYZI(NUMTGT,3) = 0
      XYZF(NUMTGT,3) = XYZI(NUMTGT,3)
      T = XYZI(1,4)
      OLDT = XYZI(1,4)-1.
      ENDTIME = XYZF(1,4)*1.0000001
      TRGPOS(1,4) = -1.
      RETURN
      END


*************************************************************


      SUBROUTINE DETPROB(POSS)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPI0B,FOPISO
```

```
      COMMON/H/FAC4,AMBN,XJAMN,IKEYJG,XXXXX
      COMMON/I/ PBBS,HOFK,THETBK,DBDOWN,THH,THV,GN
      COMMON/DET/MODEL(10)
      IF (NEXT.GT.0) GO TO 10
      PDSS=.0
      RETURN
   10 CONTINUE
C     COMPUTE SIGNAL POWER (PS) AND ENERGY (ES)
      CALL TARSIG
      WVL=300./(RC(1)+DWL(NEXT))
      S = RMODE(NEXT,3)*RC(6)*RC(6)*WVL*WVL*TARCS
      R=TRGPOS(1,4)
      RM = R*XNMTOM
      R=R*R
      R=R*R
      RMT = R*XNMTOM*XNMTOM*XNMTOM*XNMTOM
      RC(11) = S*RC(8)*RC(9)*RMODE(NEXT,8)/(RMT*FOPIOB)
      FHV(1) = 1.
      IF(RC(2).NE.1.) GO TO 50
      FH = 1.
      OAV = (PBBS-TRGPOS(1,6))
      FV = BEAM(OAV,RC(5),RC(2),1)
      FHV(1) = FH*FV
      IF(TRGPOS(1,6) .LT. RMODE(NEXT,2)) GO TO 50
      FHV(1) = AMAX1(FHV(1),DBDOWN)
   50 CONTINUE
      RC(12)=RC(11)*RMODE(NEXT,4)
      TEMPWR = RC(12)
      RC(12) = RC(12)*FHV(1)*FHV(1)
      CALL MULPTH(1,FAC)
      FAC4 = FAC*FAC*FAC*FAC
C     COMPUTE RAIN CROSS SECTION
      RNCS = 6.706E-6*TAU(NEXT)*RC(4)*RC(5)*RM *RM *WVL**(-4)
     1*ENVIR(4)**1.6
C     COMPUTE TWO WAY RAIN ATTENUATION
      RA = 10.**(-1.E-8*RM*ENVIR(4)*WVL**(-1.37))
C     COMPUTE SIGNAL ENERGY
      RC(12) = RC(12)*FAC4*AMIN1(RMODE(NEXT,12)*TAU(NEXT),1.)*RA
      RC(14) = RC(3)*RC(10)
      AMBN = RC(14)
      CALL JAM(EJ)
C     COMPUTE NOISE ENERGY TO INCLUDE RAIN ATTENUATED ENERGY AND
C        RAIN ATTENUATED SEA CLUTTER
      RC(14) = RC(14) + RC(12)*6.*RNCS/(TARCS*FAC4)*SMODE(NEXT,2)+EJ*RA
      M=IMODE(NEXT,1)*IMODE(NEXT,2)
C     COMPUTE SIGNAL TO NOISE RATIO  SNR
      SNR = RC(12)/(RC(14)*FLOAT(IMODE(NEXT,2)))
      CALL MARSWR(SNR,M,SMODE(NEXT,3),MODEL(1),PDSS)
      RETURN
      END


**************************************************************


      SUBROUTINE JAM (EJ)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
```

```
      COMMON/D/ ALPHAD,SIGZ,V,AR1,AR2,SIGC,SIG
      COMMON/H/FAC4,AMBN,XJAMN,IKEYJG,XXXXX
      COMMON/I/ PBBS,HOFK,THETBK,DBDOWN,THH,THV,GN
      EJ=0.0
      FACTOR= WVL*WVL/FOPISQ*RC(6)*RC(8)*RMODE(NEXT,8)
      HR = SHIP(3)*XNMTOM
      HT = TRGPOS(1,3)*XNMTOM
      SL = DBDOWN*DBDOWN
      AGAIN = 1
      THETB = RC(5)
      PHIB = RC(4)
      DO 20 J=1,NUMTGT
      PJ=SIGJAM(J)
      IF ( IKEYJG .EQ. 1 .AND. (ABS(TRGPOS(1,5)-TRGPOS(2,5) )) .LT.
     1         1.13*RC(4) )    PJ=0.0
      SR = TRGPOS(J,4)*XNMTOM
      IF(J.EQ.NUMTGT) GO TO 25
      IF((PJ.LE.(0.).AND.J.EQ.1).OR.SR.LE.(0.)) GO TO 20
      OAH=SMLANG(TRGPOS(1,5)-TRGPOS(J,5))
      OAV=SMLANG(TRGPOS(1,6)-TRGPOS(J,6))
      IF(RC(2).EQ.1.) OAV = (PBBS-TRGPOS(J,6))
      IF(RC(2).EQ.1.) GO TO 10
      IF (OAH.LE.(1.13*RC(4)).AND.OAV.LE.(1.13*RC(5))) GO TO 10
      FHV(J)=DBDOWN
      GO TO 15
   10 FH = BEAM(OAH,RC(4),RC(2),0)
      FV = BEAM(OAV,RC(5),RC(2),1)
      FHV(J)=FH*FV
      IF(RC(2) .EQ. 1.0 .AND. TRGPOS(J,6) .LT. RMODE(NEXT,2) .AND. FH
     1 .GT. DBDOWN) GO TO 15
      FHV(J)=AMAX1(DBDOWN,FHV(J))
   15 XJAMFA = FACTOR*PJ*FHV(J)/(SR*SR)
      CALL MULPTH(J,FAC)
      XJAMFA = XJAMFA*FAC*FAC
      EJ = EJ + XJAMFA
      XJAMN = EJ
      GO TO 20
C     CALCULATE THE EFFECT OF CLUTTER
   25 SR = TRGPOS(1,4)*XNMTOM
      IF(ENVIR(1).LT.0) GO TO 20
      DC = (1.-HR/(2.*RE))*SQRT(SR*SR - HR*HR)
      IF(DC.GE.DSTAR*XNMTOM) GO TO 20
      SINALF = HR/SR -SR/(2.*RE)
      IF(SINALF.LT.0) GO TO 20
      ALPHA = ASIN(SINALF)
      ALPHAD = ALPHA*RADIAN
      TEMP = ENVIR(1)*XKTOMS*(7.5/(ENVIR(2)*XNMTOM))**.09682
      BEAUS = ACON*(TEMP/XKTOMS)**BETA
      XFRE=RC(1)+DWL(NEXT)
      CALL CLUTSIG(XFRE,BEAUS,ALPHAD,SIGZ,POLRZ)
      V = TAU(NEXT)*CNM/(2.*COS(ALPHA))*XNMTOM
      THETT = ASIN((HT-HR)/SR - SR/(2.*RE))
      IF(RC(2).EQ.1.) THETT = PBBS-DSTAR*XNMTOM/RE
      SMTB = -SIN(THETB/2.+THETT)
      SPTB = SIN(THETB/2.-THETT)
      RADM2 = SMTB*SMTB-2.*HR/RE
      RADP2 = SPTB*SPTB-2.*HR/RE
      IF(RADM2.LT.0) GO TO 26
      RMTB = 2.*HR/(SMTB+SQRT(RADM2))
      GO TO 27
   26 RMTB = 1.E+100
   27 IF(RADP2.LT.0) GO TO 28
      RPTB = 2.*HR/(SPTB+SQRT(RADP2))
```

```
      IF(RPTB.LT.0) RPTB = 1.E+100
      GO TO 29
   28 RPTB = 1.E+100
   29 CONTINUE
      R1 = AMIN1(SR,DSTAR*XNMTOM)
      R2 = AMIN1(SR+V,DSTAR*XNMTOM)
      IF(RMTB.LT.0) RMTB = 1.E+100
      S1 = AMAX1(R1,RPTB)
      S2 = AMIN1(R2,RMTB)
      WS = .5*PHIB *(1./(S1*S1)-1./(S2*S2))
      WR = PI*(1./(R1*R1)-1./(R2*R2))
      WS = AMAX1(0.0,WS)
      AR1 = WR
      AR2 = WS
      C = TEMPWR*SR**4*SIGZ/TARCS
      SIG = C
      SIGC = C*((AGAIN-SL)*WS+SL*WR)*SUBC(NEXT)
      EJ = EJ + SIGC
   20 CONTINUE
      RETURN
      END


***********************************************************************


      SUBROUTINE CLUTSIG(XFRE,XBEAU,XANG,SIGZ,POLRZ)
      COMMON/A/ SIG0H(7,6,5),SIG0V(7,6,5),XPAR(3,7)
      DIMENSION ILIM(2,6,5),INDEX(3,2),NDX(3)
      DIMENSION PAR(3)
      DATA NDX/7,6,5/
      DATA (XPAR(1,I),I=1,7)/500.,1250.,3000.,5600.,9000.,17000.,
     1                          35000./
      DATA (XPAR(2,I),I=1,7)/1.,2.,3.,4.,5.,6.,0./
      DATA (XPAR(3,I),I=1,7)/1.,3.,10.,30.,100.,0.,0./
      DATA SIG0H/
     1             -100.,-100.,-90.,-87.,-84.,-84.,-84.,
     1             -94.,-94.,-88.,-78.,-72.,-72.,-72.,
     1             -95.,-90.,-75.,-68.,-63.,-63.,-63.,
     1             -90.,-82.,-68.,-62.,-55.,-55.,-55.,
     1             -75.,-75.,-58.,-55.,-48.,-48.,-48.,
     1             -65.,-65.,-53.,-48.,-42.,-42.,-42.,
     1             -90.,-90.,-85.,-79.,-74.,-74.,-74.,
     1             -84.,-84.,-76.,-72.,-68.,-68.,-68.,
     1             -87.,-78.,-66.,-61.,-57.,-57.,-57.,
     1             -72.,-72.,-58.,-50.,-46.,-46.,-46.,
     1             -67.,-67.,-48.,-44.,-42.,-39.,-39.,
     1             -62.,-62.,-44.,-41.,-39.,-39.,-39.,
     1             -86.,-80.,-73.,-70.,-66.,-66.,-66.,
     1             -84.,-73.,-65.,-56.,-49.,-43.,-41.,
     1             -82.,-65.,-55.,-48.,-44.,-40.,-39.,
     1             -78.,-62.,-48.,-43.,-40.,-37.,-36.,
     1             -67.,-58.,-42.,-39.,-36.,-34.,-32.,
     1            -65.,-53.,-40.,-35.,-33.,-31.,-30./
      DATA (((SIG0H(I,J,K),I=1,7),J=1,6),K=4,5)/
     1             -75.,-72.,-69.,-63.,-58.,-55.,-53.,
     1             -70.,-66.,-59.,-54.,-49.,-45.,-43.,
     1             -66.,-61.,-53.,-48.,-42.,-41.,-40.,
     1             -61.,-58.,-46.,-42.,-39.,-37.,-37.,
     1             -56.,-50.,-41.,-38.,-35.,-35.,-35.,
     1             -53.,-46.,-36.,-34.,-32.,-31.,-31.,
     1             -60.,-60.,-60.,-57.,-50.,-48.,-45.,
     1             -58.,-58.,-59.5,-53.,-51.,-42.,-34.,
```

```
     1                -55.,-55.,-51.,-46.,-43.,-35.,-33.,
     1                -50.,-48.,-46.,-40.,-37.,-33.,-31.,
     1                -48.,-45.,-38.5,-37.,-33.,-31.,-29.,
     1               -47.,-43.,-36.,-33.,-31.,-29.,-27./
      DATA SIG0V/
     1                -96.,-96.,-86.,-83.,-80.,-80.,-80.,
     1                -91.,-91.,-85.,-74.,-65.,-65.,-65.,
     1                -90.,-87.,-72.,-64.,-56.,-56.,-56.,
     1                -85.,-79.,-65.,-56.,-51.,-51.,-51.,
     1                -72.,-72.,-55.,-53.,-48.,-48.,-48.,
     1                -63.,-63.,-50.,-46.,-44.,-44.,-44.,
     1                -80.,-80.,-71.,-69.,-64.,-64.,-64.,
     1                -76.,-76.,-62.,-60.,-58.,-58.,-56.,
     1                -80.,-71.,-59.,-55.,-52.,-52.,-52.,
     1                -66.,-66.,-55.,-48.,-45.,-45.,-45.,
     1                -61.,-61.,-54.,-46.,-43.,-39.,-39.,
     1                -57.,-57.,-50.,-44.,-39.,-39.,-39.,
     1                -75.,-68.,-64.,-62.,-60.,-60.,-60.,
     1                -70.,-65.,-56.,-53.,-50.,-47.,-40.,
     1                -63.,-58.,-53.,-47.,-44.,-42.,-38.,
     1                -58.,-54.,-48.,-43.,-39.,-37.,-34.,
     1                -55.,-45.,-42.,-39.,-37.,-34.,-32.,
     1               -52.,-43.,-38.,-35.,-33.,-32.,-31./
      DATA (((SIG0V(I,J,K),I=1,7),J=1,6),K=4,5)/
     1                -65.,-64.,-64.,-60.,-56.,-52.,-46.,
     1                -60.,-53.,-52.,-49.,-45.,-43.,-41.,
     1                -55.,-53.,-49.,-45.,-41.,-39.,-37.,
     1                -43.,-43.,-43.,-40.,-38.,-36.,-34.,
     1                -38.,-38.,-38.,-36.,-35.,-33.,-31.,
     1                -38.,-38.,-35.,-33.,-31.,-31.,-30.,
     1                -45.,-45.,-47.,-48.,-49.,-45.,-44.,
     1                -40.,-40.,-42.,-44.,-42.,-40.,-38.,
     1                -35.,-37.,-38.,-39.,-36.,-34.,-33.,
     1                -34.,-34.,-34.,-34.,-32.,-32.,-31.,
     1                -30.,-31.,-31.,-32.,-31.,-29.,-29.,
     1               -25.,-28.,-28.,-28.,-26.,-26.,-26./
      DATA ILIM/2,5,2,5,1,5,1,5,2,5,2,5,
     1    2,5,2,5,1,5,2,5,2,5,2,6,
     1    1,5,1,7,1,7,1,7,1,7,1,7,
     1    1,7,1,7,1,7,1,7,1,7,1,7,
     1    2,7,2,7,1,7,1,7,1,7,1,7/
C
      ANG = POLRZ*3.14159265/180.
      PAR(1) = AMAX1(500.,AMIN1(XFRE,35000.))
      PAR(2) = AMAX1(1.,AMIN1(XBEAU,6.))
      PAR(3) = AMAX1(1.,AMIN1(10.*XANG,100.))
      DO 10 I = 1,3
      ND = NDX(I)
      DO 30 K = 1,ND
      IF(PAR(I).GT.XPAR(I,K)) GO TO 30
      INDEX(I,1) = MAX0(K-1,1)
      INDEX(I,2) = K
      GO TO 9
   30 CONTINUE
    9 IF(INDEX(I,2).NE.1) GO TO 10
      INDEX(I,2) = 2
   10 CONTINUE
C
C     CHECK FOR NONEXISTENT DATA
C
C
C     CALCULATE SIGMA ZERO
C
```

```
      IF(INDEX(1,1).EQ.INDEX(1,2).AND.(INDEX(2,1 ).EQ.INDEX(2,2)
     1.AND.INDEX(3,1).EQ.INDEX(3,2)) GO TO 50
      SIGZ = 0
      SIGZP = 0
      DO 40 I = 1,2
      DO 40 J = 1,2
      DO 40 K = 1,2
      IT1 = INDEX(1,I)
      IT2 = INDEX(2,J)
      IT3 = INDEX(3,K)
      IT1P = INDEX(1,3-I)
      IT2P = INDEX(2,3-J)
      IT3P = INDEX(3,3-K)
      SIGZP = SIGZP + SIG0V(IT1,IT2,IT3)*
     1        ABS(XPAR(1,IT1P)-PAR(1))*
     2        ABS(XPAR(2,IT2P)-PAR(2))*
     3        ABS(XPAR(3,IT3P)-PAR(3))
   40 SIGZ = SIGZ+ SIG0H(IT1,IT2,IT3)*
     1        ABS(XPAR(1,IT1P)-PAR(1))*
     2        ABS(XPAR(2,IT2P)-PAR(2))*
     3        ABS(XPAR(3,IT3P)-PAR(3))
      SIGN = 1.
      DO 80 I=1,3
      IT1 = INDEX(I,1)
      IT2 = INDEX(I,2)
      IF(XPAR(I,IT1).EQ.XPAR(I,IT2)) GO TO 80
      SIGN = SIGN*(XPAR(I,IT2)-XPAR(I,IT1))
   80 CONTINUE
      SIGZ = SIGZ/SIGN
      SIGZP = SIGZP/SIGN
      GO TO 60
   50 CONTINUE
      IT1 = INDEX(1,1)
      IT2 = INDEX(2,1)
      IT3 = INDEX(3,1)
      SIGZ = SIG0H(IT1,IT2,IT3)
      SIGZP = SIG0V(IT1,IT2,IT3)
   60 CONTINUE
      SIGZP = 10.**(SIGZP/10.)
      SIGZ= 10.**(SIGZ/10.)
      SIGZ = SQRT((SIGZ*COS(ANG))**2 + (SIGZP*SIN(ANG))**2)
      RETURN
      END


*****************************************************************


      SUBROUTINE NEWPOST(NT)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/C/ DEL(3),VEL(3),VELMAG2
      DATA NOLD/0/,TOLD/-1.E-35/
      N1=1
      IF (T.NE.TOLD) GO TO 1
      IF (NT.LE.NOLD) RETURN
      N1=NOLD+1
    1 NOLD=NT
```

```
      TOLD=T
      DO 90 J=N1,NT
      DT=(T-XYZI(J,4))/(XYZF(J,4)-XYZI(J,4))
      DOTP = 0
      DO 5 I=1,3
      TRGPOS(J,I)=XYZI(J,I)+DT*(XYZF(J,I)-XYZI(J,I))
      DEL(I)=TRGPOS(J,I)-SHIP(I)
      DOTP = DOTP + (DEL(I)*VEL(I))
    5 CONTINUE
      DSTAR=95.2*SHIP(4)
      DX=DEL(1)
      DY=DEL(2)
      DZ=DEL(3)
      D=SQRT(DX*DX+DY*DY)
      DHALF=SQRT(D)
      TRGPOS(J,4)=SQRT(DX*DX+DY*DY+DZ*DZ)
      TRGPOS(J,7) = D
      DOTP = DOTP/(SQRT(VELMAG2)*TRGPOS(J,4))
      IF (D.GT.0.000001) GO TO 10
C           THE TARGET IS NOW DIRECTLY OVERHEAD
      TRGPOS(J,6)=PIOVER2
      GO TO 20
   10 B=DHALF/95.2-SHIP(4)/DHALF
      BPRIME=TRGPOS(J,3)/D-B*B
      IF (BPRIME.LT.0.0.AND.D.GT.DSTAR) GO TO 80
C        THE TARGET HAS NOW BEEN FOUND TO BE IN SIGHT
      TRGPOS(J,6) = ATAN(BPRIME)
C       NOW COMPUTE THE TARGET HORIZONTAL ANGLE (CCW FROM EAST)
   20 IF (ABS(DX).GT.0.000001) GO TO 30
C          THE TARGET IS AT PLUS OR MINUS PI/2
      BTEMP = PIOVER2
      IF (DY.GE.0.0) GO TO 40
      BTEMP = -PIOVER2
      GO TO 40
   30 BTEMP = ATAN(DY/DX)
      IF (DX.GE.0.0) GO TO 40
      BTEMP = BTEMP+PI
   40 IF (BTEMP.GE.0.0) GO TO 50
      BTEMP = BTEMP+TWOPI
   50 TRGPOS(J,5)=BTEMP
      GO TO 90
C        THE FOLLOWING APPLIES TO AN OUT OF SIGHT TARGET
   80 TRGPOS(J,4)=-TRGPOS(J,4)
   90 CONTINUE
      RETURN
      END


C*************************************************************************


      SUBROUTINE NEXTSCN
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1      ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
   10 TMIN=1.E38
      NEXT=0
      DO 20 J=1,NSCAN
      IF (RMODE(J,9).GE.TMIN) GO TO 20
```

```
      TMIN=RMODE(J,9)
      NEXT=J
   20 CONTINUE
      T=TMIN
      J=NEXT
      RMODE(J,9)=RMODE(J,9)+RMODE(J,5)
      CALL NEWPOST(1)
      IF (TRGPOS(1,4).GT.0.0) GO TO 30
      IF (T.LE.ENDTIME) GO TO 10
      RETURN
   30 TEMP = RMODE(J,1) - RC(2)*0.63*RC(5)
      IF (TRGPOS(1,6) .GE. TEMP .AND. TRGPOS(1,6) .LE.
     1        RMODE(J,2).AND.TRGPOS(1,4).LE.RMODE(J,7)
     1        .AND.TRGPOS(1,4).GT.SMODE(J,1))RETURN
      GO TO 10
      END



******************************************************************


      FUNCTION BEAM(ALPHA,BETA,GAMMA,KEY1)
      COMMON/1/ PBBS,HOFK,THETBK,DBDOWN,THH,THV,GN
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1        ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1        ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1     ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1        ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),SUM(4,20,2)
      THETA=2.78*ALPHA/BETA
      IF(GAMMA.EQ.1..AND.KEY1.EQ.1) GO TO 40
      IF (THETA.GT.1.0E-6) GO TO 20
      SINC=1.0
      GO TO 30
   20 SINC = (SIN(THETA)/THETA)**2
      SINC=AMAX1(DBDOWN,SINC)
      IF(ABS(THETA).GT.3.14159)SINC=DBDOWN
      GO TO 30
   40 CONTINUE
      THETBK = 2.*PBBS
      HOFK = .25*(THETBK+SQRT(THETBK*THETBK+3.1*RC(5)*RC(5)))
      IF(PBBS-ALPHA.GT.HOFK) GO TO 50
      IF(ABS(THETA).GT.1.E-6) GO TO 55
      SINC = 1.
      GO TO 30
   55 SINC = (SIN(THETA)/THETA)**2
      GO TO 30
   50 CONTINUE
      IF(PBBS-ALPHA .LT. RMODE(NEXT,2)) GO TO 60
      SINC = DBDOWN
      GO TO 30
   60 CONTINUE
      THETA = 2.783*(HOFK-PBBS)/BETA
      SINC = (SIN(THETA)/THETA*SIN(HOFK)/SIN(PBBS-ALPHA))**2
   30 BEAM = SINC
   65 CONTINUE
      RETURN
      END



******************************************************************


      FUNCTION SMLANG (ANGLE)
```

```
      PI = 3.1415926536
      TWOPI = 2.*PI
      A=ABS(ANGLE)
   10 IF (A.LT.TWOPI) GO TO 20
      A=A-TWOPI
      GO TO 10
   20 IF (A.LT.PI) GO TO 30
      A=TWOPI-A
   30 SMLANG=A
      RETURN
      END

*****************************************************************************


      SUBROUTINE MATCH
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1      ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1      ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1  ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1      ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1      DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FCPIOB,FCPISO
      COMMON/C/ A(3),B(3),BMAG2
C    MATCH SCANTIME VALUES WITH TIMES THAT THE TARGET
C    COMES WITHIN THE SCOPE LIMITS.
      DO 10 K=1,3
      A(K)=XYZI(1,K)-SHIP(K)
      B(K)=XYZF(1,K)-XYZI(1,K)
   10 CONTINUE
      AMAG2=A(1)*A(1)+A(2)*A(2)+A(3)*A(3)
      BMAG2=B(1)*B(1)+B(2)*B(2)+B(3)*B(3)
      ADOTB=A(1)*B(1)+A(2)*B(2)+A(3)*B(3)
      ADOTB2=ADOTB*ADOTB
      DO 80 I=1,NSCAN
      RMODE(I,9)=RMODE(I,6)+XYZI(1,4)
      R2=RMODE(I,7)*RMODE(I,7)
      DISC=ADOTB2-BMAG2*(AMAG2-R2)
      IF (DISC.GE.0.0) GO TO 60
C           THIS APPLIES TO A TARGET WHICH NEVER GETS WITHIN RANGE
   50 RMODE(I,9)=ENDTIME+1.0
      GO TO 80
   60 TERM=SQRT(DISC)
      UMINUS=-1.0*(ADOTB+TERM)/BMAG2
      UPLUS=(TERM-ADOTB)/BMAG2
      WMINUS=AMAX1(UMINUS,0.0)
      WPLUS=AMIN1(UPLUS,1.0)
      IF (WMINUS.GT.WPLUS) GO TO 50
      XSCAN=XYZI(1,4)+WMINUS*(XYZF(1,4)-XYZI(1,4))
      IF (RMODE(I,9).GE.XSCAN) GO TO 80
      XN=AINT((XSCAN-RMODE(I,9))/RMODE(I,5))
      RMODE(I,9)=RMODE(I,9)+XN*RMODE(I,5)
      IF (XSCAN.EQ.RMODE(I,9)) GO TO 80
      RMODE(I,9)=RMODE(I,9)+RMODE(I,5)
   80 CONTINUE
      RETURN
      END


*****************************************************************************

```

```
      SUBROUTINE TARSIG
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1      ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1      ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1       ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1      ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1      DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      DIMENSION TEMP(4)
      COMMON/E/ A(4)
      TEMP(1) = DOTP
      DO 10 I = 2,4
   10 TEMP(I) = 2.*TEMP(I-1)**2-1.
      TARCS = A(4) + A(3)*TEMP(4) + A(2)*TEMP(3) - A(1)*TEMP(2)
      RETURN
      END


*************************************************************


      SUBROUTINE INTER(DELT,A,AA,B,BB,C,CC)
      ADB = ALOG(A) - DELT*ALOG(AA)
      BDB = ALOG(B) - DELT*ALOG(BB)
      CDB = (BDB-ADB)*(CC-AA)/(BB-AA) + ADB
      C = CC**DELT*EXP(CDB)
      RETURN
      END


*************************************************************


      SUBROUTINE UFUN(Z,CA,CANS)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1      ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1      ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1       ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1      ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1      DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMPLEX CA,CANS,CI,CXP3,CZ,CA1,CZ3,CA13,CFI,CFIP,CGI,CGIP
     1     ,CF,CFP,CG,CGP,CH2,CH2P,CDZ,CDA,CZ4,CA14,CCE,CCEA
     1     ,CC,CCP,CAIR,CAIRP,CFM,CFPM,CGM,CGPM,COLDA
      IF(ISWIT.EQ.1) GO TO 5
      CI = (0.0,1.)
      XC1 = .3550280539
      XC2 = .2588194038
      RTPI = SQRT(PI)
      CXP3 = CEXP(CI*PI/3.)
      CA1 = CA*CXP3
      OLDZ = 0
    5 CONTINUE
      IF(Z.EQ.OLDZ) GO TO 80
      OLDZ = Z
      CZ = (Z+CA)*CXP3
      IKEY = 0
      IF(CABS(CZ).GT.3.) IKEY = IKEY+1
      IF(CABS(CA1).GT.3.) IKEY = IKEY+2
      IF(IKEY.EQ.3) GO TO 10
      IF(ISWIT.EQ.1.AND.IKEY.EQ.1) GO TO 10
      CZ3 = CZ*CZ*CZ
      CA13 = CA1*CA1*CA1
```

```
      CFI = 1
      CFIP = 0
      CGI = CA1
      CGIP = 1.
      CFM = 1.
      CFPM = 1./CA1
      CGM = CZ
      CGPM = 1
      XMUL1 = 1
      XMUL2 = 1
      I = 0
   20 I = I+1
      XMUL1 = XMUL1/(3*I*(3*I-1))
      XMUL2 = XMUL2/((3*I+1)*3*I)
      CFM = CFM*CZ3
      CFPM = CFPM*CA13*(3*I)/(MAX0(1,3*I-3))

      CGM = CGM*CZ3
      CGPM = CGPM*CA13*(3*I+1)/(3*I-2)
      CF = XMUL1*CFM
      CFP = XMUL1*CFPM
      REA1 = REAL(CF)
      COM1 = AIMAG(CF)
      REA2 = REAL(CFP)
      COM2 = AIMAG(CFP)
      CG = XMUL2*CGM
      CGP = XMUL2*CGPM
      CFI = CFI+CF
      CFIP = CFIP+CFP
      CGI = CGI+CG
      CGIP = CGIP+CGP
      IKEY1 = IKEY+1
      REA = REAL(CFI)
      COM = AIMAG(CFI)
       GO TO (22,23,24) IKEY1
   22 IF(CABS(XC1*CF-XC2*CG).LT..005*CABS(XC1*CFI-XC2*CGI).AND.
     1   CABS(XC1*CFP-XC2*CGP).LT..005*CABS(XC1*CFIP-XC2*CGIP))
     1   GO TO 21
      GO TO 20
   23 IF(CABS(XC1*CFP-XC2*CGP).LT..005*CABS(XC1*CFIP-XC2*CGIP))
     1   GO TO 21
      GO TO 20
   24 IF(CABS(XC1*CF-XC2*CG).LT..005*CABS(XC1*CFI-XC2*CGI))
     1   GO TO 21
      GO TO 20
   21 CONTINUE
      IF(IKEY.EQ.1) GO TO 25
      CH2 = CI*3.02617*(XC1*CFI-XC2*CGI)
   25 IF(IKEY.LT.2) CH2P = CI*3.02617*(XC1*CFIP-XC2*CGIP)*CXP3
      IF(IKEY.EQ.0) GO TO 70
      IF(ISWIT.EQ.1.AND.IKEY.EQ.2) GO TO 70
   10 CONTINUE
      CDZ = 2./3.*CEXP(1.5*CLOG(CZ))
      CDA = 2./3.*CEXP(1.5*CLOG(CA1))
      CZ4 = 1./CSQRT(CSQRT(CZ))
      CA14 = CSQRT(CSQRT(CA1))
      CCE = CEXP(-CDZ)
      CCEA = CEXP(-CDA)
      CC = .5*CZ4*CCE/RTPI
      CCP = -.5*CA14*CCEA/RTPI
      XMUL1 = 1.
      CAIR = 1
      CAIRP = 1
      CFM = 1.
```

```
      CFPM = 1.
      I = 0
   40 I = I+1
      K = 6*I-1
      DO 50 J = 1,3
      XMUL1 = XMUL1*K
      K = K-2
   50 CONTINUE
      XMUL1 = XMUL1/(216*I*(2*I-1))
      CFM = CFM/CDZ
      CF = XMUL1*CFM
      CFPM = CFPM*((6*I+1)/(1-6*I))/CDA
      CFP = XMUL1*CFPM
      CAIR = CAIR+(-1)**I*CF
      CAIRP = CAIRP+(-1)**I*CFP
      GO TO (51,52,53) IKEY
   51 IF(CABS(CF).LT..002) GO TO 60
      GO TO 40
   52 IF(CABS(CFP).LT..002) GO TO 60
      GO TO 40
   53 IF(CABS(CF).LT..002.AND.CABS(CFP).LT..002) GO TO 60
      GO TO 40
   60 CONTINUE
      IF(IKEY.NE.2) CH2 = CI*3.02617*CC*CAIR
      IF(IKEY.GE.2) CH2P = CI*3.02617*CCP*CAIRP*CXP3
   70 CONTINUE
      CANS = CI*CH2/CH2P
      COLDA = CANS
      RETURN
   80 CANS = COLDA
      RETURN
      END


************************************************************************


      SUBROUTINE RNGCEN(DELT,SR1,SR2,R1,R2,S)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1       ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1       ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1       ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1       ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTONS,XNMTCM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/F/HR,HT
      I = 0
   10 CONTINUE
      ASS = RE*S*S
      V1 = SQRT(1.+2.*HR/ASS)
      V2 = SQRT(1.+2.*HT/ASS)
      V = (1.+V1)/HR + (1.+V2)/HT
      F = 4.*S/V - DELT
      FP = 4.*V + 8./ASS * (1./V1+1./V2)
      FP = FP/(V*V)
      S = S-F/FP
      IF(ABS(F).LT.DELT/10.) GO TO 20
      IF(I.GT.10) GO TO 20
      I = I+1
      GO TO 10
   20 CONTINUE
      ASS = RE*S*S
      T = 2.*HR
```

```
      R1 = T/(S+SQRT((ASS+T)/RE))
      T = 2.*HT
      R2 = T/(S+SQRT((ASS+T)/RE))
      SR1 = SQRT((R1*R1-HR*HR)/(1.+HR/RE))
      SR2 = SQRT((R2*R2-HT*HT)/(1.+HT/RE))
      RETURN
      END


************************************************************


      SUBROUTINE GAIN(IKEY,ITAR,GAINR)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1       ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1       ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1        ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1       ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/G/ SRTAR,SINPSI,COSPSI
      COMMON/I/ PBBS,HOFK,THETBK,DBDOWN,THH,THV,GN
      IF(IKEY.EQ.1) GO TO 10
      GAINR = SQRT(FHV(ITAR))
      RETURN
   10 CONTINUE
      OAH = SMLANG(TRGPOS(1,5)-TRGPOS(ITAR,5))
      ALFV = (2.*SRTAR*SINPSI*COSPSI/(TRGPOS(ITAR,4)*XNMTOM))
      IF(ALFV.LE.1.)GO TO 30
      IF(ALFV.LE.1.01)GO TO 40
   40 ALFV=1.
   30 CONTINUE
      ALFV=ASIN(ALFV)
      ALFV = TRGPOS(ITAR,6)-ALFV
      OAV = SMLANG(TRGPOS(1,6)-ALFV)
      IF(RC(2).EQ.1.) OAV = (PBBS-ALFV)
      IF(RC(2).EQ.1.) GO TO 20
      IF(OAH.LE.(1.13*RC(4)).AND.OAV.LE.(1.13*RC(5))) GO TO 20
      GAINR = RC(7)
      RETURN
   20 FH = BEAM(OAH,RC(4),RC(2),0)
      FV = BEAM(OAV,RC(5),RC(2),1)
      GAINR = FH*FV
      GAINR = AMAX1(RC(7),SQRT(GAINR))
      RETURN
      END


************************************************************


      SUBROUTINE MULPTH(ITAR,FAC)
      COMMON NSCAN,NEXT,NUMTGT,T,OLDT,ENDTIME,SMODE(15,10)
     1       ,PI,PIOVER2,TWOPI,RADIAN,TAU(15),DSTAR,DWL(15)
     1       ,XYZI(10,4),XYZF(10,4),TRGPOS(10,7),SIGJAM(10)
     1        ,SIGTAR(10,3),FHV(10),SIGMAH,ISWIT,TEMPWR
     1       ,SHIP(4),RC(15),RMODE(15,12),IMODE(15,2),CUM(4000,2)
      COMMON/B/ ENVIR(10),SUBC(15),RE,CNM,CCM,ACON,BETA,
     1        DOTP,POLRZ,IKEYF,XKTOMS,XNMTOM,TARCS,WVL,FOPIOB,FOPISO
      COMMON/F/ HR,HT
      COMMON/G/SRTAR,SINPSI,COSPSI
      COMPLEX CI,CTEMY,CP,CA,CU1,CU2,CTEMW,CGAMV,CGAMH
     1      ,CGAM
      FAC = 1.
```

```
      IF(ENVIR(3) .EQ. 0.) RETURN
      IF(ISWIT.EQ.1) GO TO 5
      XNZERO = 1.000313
      HR = SHIP(3)*XNMTOM
      RAR = HR/RE
      CHR = HR*(1.-RAR)
      XKPAR = 2.*PI/WVL
      XFRE=RC(1)+DWL(NEXT)
      IF(XFRE.GT.1500.) GO TO 6
      EPS1 = 80.
      SIG1 = 4.3
      GO TO 8
    6 IF(XFRE.GT.3000.) GO TO 7
      EPS1 = 80.-0.00733*(XFRE-1500.)
      SIG1 = 4.3+0.00148*(XFRE-1500.)
      GO TO 8
    7 EPS1 = 69-0.0005714*(XFRE-3000.)
      SIG1 = 6.52+0.001314*(XFRE-3000.)
    8 CONTINUE
      CI = (0.0,1.)
      CTEMY = CMPLX(EPS1,-60.*WVL*SIG1)
      XKZERO = XKPAR*XNZERO
      H = (2.*XKZERO*XKZERO/RE)**(-1./3.)
      XL = 2.*(4.*XKZERO/(RE*RE))**(-1./3.)
      Z1 =HR/H
      CP = CSQRT(CTEMY - 1.)
      IF (REAL(CP) .LT. 0.) CP = -1.0*CP
      CP = CP*CI*XKZERO*(COS(POLRZ/RADIAN)
     1     + SIN(POLRZ/RADIAN)/CTEMY)
      CA = 2.3381*CEXP(CI*2.*PI/3.) + 1./(H*CP)
      XIMCA = AIMAG(CA)
      CALL UFUN(Z1,CA,CU1)
      ISWIT = 1
    5 CONTINUE
      IKEY = -1
      GRRT = TRGPOS(ITAR,7)*XNMTOM
   20 CONTINUE
      HT = TRGPOS(ITAR,3)*XNMTOM
      HSUM = HR+HT
      RAT = HT/RE
      CHT = HT*(1.-RAT)
      TEMP = SQRT(4./3.*(RE*(CHR+CHT)+GRRT*GRRT/4.))
      TEMPHI = ACOS(2.*RE*GRRT*ABS(CHR-CHT)/(TEMP*TEMP*TEMP))
      GRRAD = GRRT/2.+SIGN(1.,CHR-CHT)*TEMP*COS((TEMPHI+PI)/3.)
      GRTAR = GRRT-GRRAD
      SRRAD = SQRT(HR*HR+(1.+RAR)*GRRAD*GRRAD)
      SRTAR = SQRT(HT*HT+(1.+RAT)*GRTAR*GRTAR)
      TANPSI = CHR/GRRAD - GRRAD/(2.*RE)
      IF(TANPSI.LE.0) GO TO 40
      COSPSI = GRRAD/SRRAD*(1.+RAR)
      SINPSI = GRRAD/SRRAD*(TANPSI+HR/GRRAD*RAR)
   50 CONTINUE
      RREF = SRRAD+SRTAR
      PRREF = SRRAD*SRTAR
      TEMW = 2.*PRREF/RREF
      DISP = SQRT((1.-HSUM/RE)*RREF/GRRT
     1   *COSPSI/(1.+TEMW/(RE*SINPSI)))
      TEMG = (2.*SINPSI/RREF)**2*PRREF
      PTHDIF = RREF*TEMG/(1.+SQRT(1.-TEMG))
      IF(IKEY.EQ.1) GO TO 30
      IF(PTHDIF.GE.WVL/4.) GO TO 30
      IKEY = 1
      CRITR = GRRT
```

```
      XNAT = CRITR/XL
      SINPSI = 2.*WVL/4.*HSUM/(WVL/4.*WVL/4.+4.*HR*HT)
      CALL RNGCEN(WVL/4.,GRRAD,GRTAR,SRRAD,SRTAR,SINPSI)
      GRRT = GRRAD + GRTAR
      CRITR = GRRT
      COSPSI = SQRT(1.-SINPSI*SINPSI)
      XNAT1 = CRITR/XL
      GO TO 50
   30 CONTINUE
      CTEMW = CSQRT(CTEMY-COSPSI*COSPSI)
      IF(REAL(CTEMW) .LT. 0.)  CTEMW = -1.0*CTEMW
      CGAMV = (CTEMY*SINPSI-CTEMW)/(CTEMY*SINPSI+CTEMW)
      CGAMH = (SINPSI-CTEMW)/(SINPSI+CTEMW)
      CGAM =       .5*CLOG((CGAMH*COS(POLRZ/RADIAN))**2
     1         +(CGAMV*SIN(POLRZ/RADIAN))**2)
      CGAMI=AIMAG(CGAM)
      IF ( CGAMI .GT. 0.0 ) GOTO 51
      CGAM=CMPLX( REAL(CGAM),3.1416+CGAMI)
   51 CGAM=CEXP( CGAM )
      RHOREF = CABS(CGAM)
      PHIREF = AIMAG(CLOG(CGAM))
      TEMG1 = (2.*XKPAR*SIGMAH*SINPSI)**2
      XMUR = EXP(-TEMG1/2.)
      CALL GAIN(1,ITAR,GV)
      CALL GAIN(0,ITAR,GD)
      FAC1 = CABS(1.+GV/GD*RHOREF*DISP*XMUR*CEXP(CI*(-XKZERO*
     1      PTHDIF-PHIREF)))
      IF(IKEY.EQ.-1) GO TO 10
      Z2 = HT/H
      XNAT2 = 2.*(SQRT(2.*RE*CHT) + SQRT(2.*RE*CHR))/XL
      CALL UFUN(Z2,CA,CU2)
      FAC2 = 2.*SQRT(PI*XNAT2)*EXP(-XIMCA*XNAT2)*CABS(CU1*CU2)
      CALL INTER(0,FAC1,XNAT1,FAC2,XNAT2,FAC,XNAT)
      RETURN
   10 FAC = FAC1
      RETURN
   40 CONTINUE
      FAC = 1.E-20
      RETURN
      END


*****************************************************************


      SUBROUTINE MARSWR(SNR,N,FA,KASE,PN)
C
C     INPUTS ARE -- SNR, SIGNAL-TO-NOISE RATIO --
C     N, NUMBER OF PULSES INTEGRATED --
C     FA, FALSE ALARM PROBABILITY, EXPRESSED AS ABSOLUTE VALUE OF POWER
C     OF TEN (E.G.. FA = 8. MEANS 10.**(-8.) FALSE ALARM PROBABILITY --
C     KASE, SWERLING FLUCTUATION MODEL, WITH KASE = 0 FOR NONFLUCTUATING
C     OUTPUT PN IS PROBABILITY OF DETECTION
C
C     BASED ON PROGRAM WRITTEN ATJHU APPLIED PHYSICS LABORATORY, NAMED
C     SUBROUTINE MARCUM. MODIFIED AT NRL BY L. V. BLAKE. THIS VERSION
C     DATED APRIL 1971
C     APL VERSION DEFINED FA AS FALSE ALARM NUMBER (MARCUM CONCEPT).
C     NRL MOD CHANGED THIS TO FALSE ALARM PROBABILITY (AS DEFINED ABOVE)
C     SOME OTHER CHANGES ALSO.
C
```

```
      DOUBLE PRECISION ENPR,YBPR,GAMPR,PYB,H,Y0,E0,Y1,E1,STEP,YB
      DOUBLE PRECISION DGAM, DEVAL, SUMLOG, SUML, FAN, EN
C
C     COMPUTE MARCUM-SWERLING DETECTION PROBABILITIES
C
C
      MODE=1
C     IF MODE IS 1, CONVERT FA  TO MEAN EXPONENT OF FALSE-ALARM
C     PROBABILITY RATHER THAN MARCUM FALSE-ALARM NUMBER
C
      IF (MODE) 800, 800, 900
  900 FAN =DLOG10(DLOG(.5D0)/DLOG(1.-(10.D0)**(-FA)))
      GO TO 905
  800 FAN = FA
C
C     TEST INPUTS
C
  905 IF(N) 99,99,2
    2 IF(FA)99,99,3
    3 IF(KASE) 99,4,4
    4 IF(KASE-4) 5,5,99
C
C     ESTIMATE BIAS LEVEL
C
    5 ENPR = 0.
    6 ENPR = FAN
      EN = N
      YBPR = 0.
      IF (NPREV .EQ. N .AND. FAPREV .EQ. FA) GO TO 777
      IF(N-12) 7,7,8
    7 YBPR = EN*(1.+2.2*ENPR/EN**((2.D0/3.D0)+.015*ENPR))
      GO TO 11
    8 YBPR = EN*(1.+1.3*ENPR/EN**(.5+.011*ENPR))
C
C     COMPUTE BIAS LEVEL
C
   11 ENPR = 10.**ENPR
      GAMPR = DGAM(YBPR,N-1)
      PYB = .5**(1./ENPR)
      SUML = SUMLOG(N-1)
      IF(GAMPR-PYB) 10,12,12
   10 H = .01
      GO TO 14
   12 H = -.01
   14 Y0 = YBPR
      E0 = DEVAL(Y0,N-1,SUML)
   16 Y1 = Y0+H
      E1 = DEVAL(Y1,N-1,SUML)
      STEP = GAMPR + H*(E0+E1)/2.
      IF(DSIGN(1.D0,STEP-PYB)-DSIGN(1.D0,H)) 18,20,18
   18 Y0 = Y1
      E0 = E1
      GAMPR = STEP
      GO TO 16
   20 IF(H) 22,24,24
   22 YB = Y1 - H*(PYB-STEP)/(GAMPR-STEP)
      GO TO 30
   24 YB = Y0 + H*(PYB-GAMPR)/(STEP-GAMPR)
   30 BIAS = YB
  777 YB = BIAS
      NPREV = N
      FAPREV = FA
C
```

```
C      SELECT M-S CASE
C
       X = SNR
       K = KASE+1
       GO TO (100,200,300,400,500), K
C
C      CASE 0
C
  100  SUM = 0.
C      PATCH TO AVOID EXP OVERFLOW
       IF(SNR .LT. 60.) GO TO 101
       PN = 1.
       RETURN
  101  CONTINUE
       P = EN*X
       IF(YB-P-EN) 150,102,102
  102  KS = -(EN+1.)/2. + SQRT(((EN-1.)/2.)**2+P*YB)
       KS = MAX0(KS,0)
       GS = 1.-GAM(YB,KS+N-1,TN)
       TS = EVAL(P,KS)*GS
       G = GS
       K = KS
       TERM = TS
       TL = TN
  110  TEMP = SUM+TERM
       IF(SUM-TEMP) 112,116,116
  112  SUM = TEMP
       IF(K) 116,116,114
  114  TERM = TERM*FLOAT(K)*(G-TL)/(P*G)
       G = G-TL
       K = K-1
       TL = TL*FLOAT(K+N)/YB
       GO TO 110
  116  TL = TN*YB/FLOAT(KS+N)
       K = KS+1
       G = GS+TL
       TERM = TS*P*G/(GS*FLOAT(K))
  120  TEMP = SUM+TERM
       IF(SUM-TEMP) 122,190,190
  122  SUM = TEMP
       TL = TL*YB/FLOAT(K+N)
       K = K+1
       TERM = TERM*P*(G+TL)/(G*FLOAT(K))
       G = G+TL
       GO TO 120
  150  KS = -1. - EN/2. + SQRT(EN**2/4.+P*YB)
       KS =MAX0(KS,0)
       GS = GAM(YB,KS+N-1,TN)
       IF(GS) 174,174,155
  155  TS = EVAL(P,KS)*GS
       G = GS
       TERM = TS
       K = KS
       TL = TN
  160  TEMP = SUM+TERM
       IF(SUM-TEMP) 162,166,166
  162  SUM = TEMP
       IF(K) 166,166,164
  164  TERM = TERM*FLOAT(K)*(G+TL)/(P*G)
       G = G+TL
       TL = TL*FLOAT(K+N-1)/YB
       K = K-1
       GO TO 160
```

```
      166 TL = TN*YB/FLOAT(KS+N)
          K = KS+1
          G = GS-TL
          TERM = TS*P*G/(GS*FLOAT(K))
      170 TEMP = SUM + TERM
          IF(SUM-TEMP) 172,174,174
      172 SUM = TEMP
          TL = TL*YB/FLOAT(K+N)
          TERM = TERM*P*(G-TL)/(G*FLOAT(K+1))
          G = G-TL
          K = K+1
          GO TO 170
      174 SUM = 1.-SUM
      190 PN = SUM
          GO TO 90
C
C         CASE 1
C
      200 IF(N-1) 210,210,220
      210 PN = EXP(-YB/(1.+X))
          GO TO 90
      220 TEMP = 1. + 1./(EN*X)
          PN = 1. - GAM(YB,N-2,DUM) + EXP((EN-1.)*ALOG(TEMP)-YB/(1.+EN*X))
         1                       *GAM(YB/TEMP,N-2,DUM)
          GO TO 90
C
C         CASE 2
C
      300 IF(N-1) 310,310,320
      310 PN = EXP(-YB/(1.+X))
          GO TO 90
      320 PN = 1. - GAM(YB/(1.+X),N-1,DUM)
          GO TO 90
C
C         CASE 3
C
      400 IF(N-2) 410,420,430
      410 PN = (1.+2.*X*YB/(X+2.)**2)*EXP(-2.*YB/(2.+X))
          GO TO 90
      420 PN = (1.+YB/(1.+X))*EXP(-YB/(1.+X))
          GO TO 90
      430 C = 2./(2.+EN*X)
          D = 1.-C
          IF(YB*D-EN) 440,450,450
      440 SUM = 0.
          TERM = 1.
          J = N
      442 TEMP = SUM+TERM
          IF(SUM-TEMP) 444,446,446
      444 SUM = TEMP
          TERM = TERM*YB*D/FLOAT(J)
          J = J+1
          GO TO 442
      446 PN = 1. - GAM(YB,N-2,DUM) + C*YB*EVAL(YB,N-2)
         1         + D*EVAL(YB,N-1)*(1.+C*YB-(EN-2.)*C/D)*SUM
          GO TO 90
      450 PN = 1. - GAM(YB,N-3,DUM) + YB*EVAL(YB,N-3)*C/D
         1         + EXP(-C*YB-(EN-2.)*ALOG(D))*(1.+C*YB-(EN-2.)*C/D)
         2         *GAM(YB*D,N-3,DUM)
          GO TO 90
C
C         CASE 4
C
```

```
      500 SUM = 0.
          C = 2./(2.+X)
          D = 1.-C
          O = C/D
          P = C*YB
          KS = (3.*EN+(YB*D))/2.-SQRT((EN-1.+(YB*D))**2/4.+(YB*D)*(EN+1.))
          KS = MIN0(KS,N)
          KS = MAX0(KS,0)
          K = KS
          J = N-KS
          FKS = KS
          K = MIN0(KS,N)
          IF(YB-EN*(1.+D)) 550,501,501
      501 GS = 1. - GAM(P,2*N-1-KS,TN)
          IF(GS) 526,526,502
      502 TS = EXP(FKS*ALOG(C)+(EN-FKS)*ALOG(D)+SUMLOG(N)-SUMLOG(KS)
         1            -SUMLOG(J)+ALOG(GS))
          G = GS
          TERM = TS
          TL = TN
      510 TEMP = SUM+TERM
          IF(SUM-TEMP) 512,516,516
      512 SUM = TEMP
          IF(K) 516,516,514
      514 TL = TL*P/FLOAT(2*N-K)
          TERM = TERM*FLOAT(K)*(G+TL)/(O*FLOAT(N-K+1)*G)
          G = G+TL
          K = K-1
          GO TO 510
      516 IF(KS-N) 518,526,526
      518 TERM = TS*O*FLOAT(N-KS)*(GS-TN)/(FLOAT(KS+1)*GS)
          G = GS-TN
          TL = TN*FLOAT(2*N-1-KS)/P
          K = KS+1
      520 TEMP = SUM+TERM
          IF(SUM-TEMP) 522,526,526
      522 SUM = TEMP
          IF(K-N) 524,526,526
      524 TERM = TERM*O*FLOAT(N-K)*(G-TL)/(FLOAT(K+1)*G)
          G = G-TL
          TL = TL*FLOAT(2*N-1-K)/P
          K = K+1
          GO TO 520
      526 PN = SUM
          GO TO 90
      550 GS = GAM(P,2*N-1-KS,TN)
          IF(GS) 576,576,552
      552 TS = EXP(FKS*ALOG(C)+(EN-FKS)*ALOG(D)+SUMLOG(N)-SUMLOG(KS)
         1            -SUMLOG(J)+ALOG(GS))
          G = GS
          TERM = TS
          TL = TN
      560 TEMP = SUM+TERM
          IF(SUM-TEMP) 562,566,566
      562 SUM = TEMP
          IF(K) 566,566,564
      564 TL = TL*P/FLOAT(2*N-K)
          TERM = TERM*FLOAT(K)*(G-TL)/(O*FLOAT(N-K+1)*G)
          G = G-TL
          K = K-1
          GO TO 560
      566 IF(KS-N) 568,576,576
      568 TERM = TS*O*FLOAT(N-KS)*(GS+TN)/(FLOAT(KS+1)*GS)
```

```
      G = GS+TN
      TL = TN*FLOAT(2*N-1-KS)/P
      K = KS+1
  570 TEMP = SUM+TERM
      IF(SUM-TEMP) 572,576,576
  572 SUM = TEMP
      IF(K-N) 574,576,576
  574 TERM = TERM*Q*FLOAT(N-K)*(G+TL)/(FLOAT(K+1)*G)
      G = G+TL
      TL = TL*FLOAT(2*N-1-K)/P
      K = K+1
      GO TO 570
  576 PN = 1.-SUM
      GO TO 90
C
C     SET PROBABILITY
C
   90 IF(PN) 91,94,92
   91 PN = 0.
      GO TO 94
   92 IF(PN-1.) 94,94,93
   93 PN = 1.
   94 RETURN
C
C     ERROR MESSAGE FOR BAD INPUTS
C
   99 TYPE 9,  N,FA ,SNR,KASE
    9 FORMAT (1H0 /50H UNREASONABLE CALL SEQUENCE TO MARCUM, ZERO RESULT
     1               7HS GIVEN //4H N = I8,5X,5HFA  = E16.8,5X,5HSNR =
     2             E16.8,5X,6HKASE = I8)
      PN = 0.
      BIAS = 0.
      RETURN
      END


***************************************************************


      FUNCTION DGAM(B,N)
      DOUBLE PRECISION  SUM,  TERM,TEMP,FJ,DGAM, DEVAL, B, SUML,SUMLOG
C          INTEGRAL = 1-(SUM, J=0 TO N, OF EXP(J*ALOG(B)-B-ALOG(NFAC)
      SUM = 0.
      K = B
      IF(K-N) 100,200,200
  100 J = N+1
      SUML = SUMLOG(J)
      TERM = DEVAL(B,J,SUML)
   10 TEMP = SUM+TERM
      IF(SUM-TEMP) 15,20,20
   15 SUM = TEMP
      J = J+1
      FJ = J
      TERM = TERM*B/FJ
      GO TO 10
   20 DGAM = SUM
      RETURN
  200 J = N
      SUML = SUMLOG(J)
      TERM = DEVAL(B,J,SUML)
   30 TEMP = SUM+TERM
      IF(SUM-TEMP) 35,40,40
```

```
      IF(J-1) 40,36,36
   36 FJ = J
      TERM = TERM*FJ/B
      J = J-1
      GO TO 30
   40 DGAM = 1.-SUM
      RETURN
      END


****************************************************************


      FUNCTION DEVAL (Y,N,SUML)
      DOUBLE PRECISION  XPON,EN,DEVAL,  Y,SUML
      XPON = -Y
      IF(N) 20,20,10
   10 EN = N
      XPON = XPON+EN*DLOG(Y)-SUML
   20 DEVAL = DEXP(XPON)
      RETURN
      END


****************************************************************


      FUNCTION GAM(B,N,TN)
C                    SINGLE PRECISION VERSION OF DGAM
      SUM = 0.
      K = B
      IF(K-N) 100,200,200
  100 J = N+1
      TERM = EVAL(B,J)
      TN = TERM*FLOAT(J)/B
   10 TEMP  = SUM+TERM
      IF(SUM-TEMP) 15,20,20
   15 SUM = TEMP
      J = J+1
      FJ = J
      TERM = TERM*B/FJ
      GO TO 10
   20 GAM = SUM
      RETURN
  200 J = N
      TERM = EVAL(B,J)
      TN = TERM
   30 TEMP = SUM+TERM
      IF(SUM-TEMP) 35,40,40
   35 SUM = TEMP
      IF(J-1) 40,36,36
   36 FJ = J
      TERM = TERM*FJ/B
      J = J-1
      GO TO 30
   40 GAM = 1.-SUM
      RETURN
      END


****************************************************************
```

```
      FUNCTION EVAL(Y,N)
      XPON = -Y
      IF(N) 20,20,10
   10 EN = N
      XPON = XPON+EN*ALOG(Y)-SUMLOG(N)
   20 EVAL = EXP(XPON)
      RETURN
      END


*****************************************************************


      FUNCTION SUMLOG(N)
      DOUBLE PRECISION A, B, SUMLOG
      DIMENSION A(1000)
      DATA DUMA/0./,DUMB/0./
      NMAX=1000
      IF(DUMA-DUMB) 20,10,20
   10 DUMA = 1.
      DUMB = 0.
      NLAST = 1
      A(1) = 0.
   20 NN = IABS(N)
      IF(NN-1) 30,30,40
   30 SUMLOG = 0.
      RETURN
   40 IF(NN-NLAST) 50,50,60
   50 SUMLOG = A(NN)
      RETURN
   60 K = NLAST+1
      IF(NN-NMAX) 70,70,80
   70 DO 72 I=K,NN
   72 A(I) = A(I-1) + DLOG(FLOAT(I)*1.D0)
      NLAST = NN
      GO TO 50
   80 IF(NLAST-NMAX) 82,90,90
   82 DO 84 I=K,NMAX
   84 A(I) = A(I-1) + DLOG(FLOAT(I)*1.D0)
      NLAST = NMAX
   90 B = A(NMAX)
      K = NMAX+1
      DO 92 I=K,NN
   92 B = B + DLOG(FLOAT(I)*1.D0)
      SUMLOG = B
      RETURN
      END


.K/F
JOB 7, USER [307,747]  LOGGED OFF TTY14     1058  8-JUL-75
SAVED ALL FILES (510 BLOCKS)
RUNTIME 54.60 SEC
```

Appendix B

SUBROUTINE MTIMOD

W. B. Gordon

B.1 INTRODUCTION

A Moving Target Indicator (MTI) system is a signal processor that distinguishes moving targets from clutter by virtue of the differences in their spectra. The simplest MTI processor, the single delay-line canceler, obtains the difference between two successive echoes from the same location; reflections from stationary objects cancel, whereas reflections from moving targets produce fluctuating signals. Thus MTI canceler systems maximize signal-to-clutter ratios only for highly correlated interference; uncorrelated interference, such as receiver noise, is not affected by the MTI processor.

MTIMOD, the MTI processor model described here, is based up Chapters 6 and 9 of *Radar Design Principles*, by Nathanson,* but has been generalized by permitting the use of clutter-rejection filters not discussed by Nathanson. It is assumed that the radar system contains a matched filter for the individual pulses to enhance the signal-to-noise ratio and that the clutter-to-noise ratio is large.

MTIMOD predicts MTI performance in detecting a target located in a background of precipitation and chaff by calculating the clutter power input to each of one or more specified clutter-rejection filters, the clutter power output from each clutter-rejection filter, and the signal power output from each clutter-rejection filter. The model includes the following often neglected aspects of clutter analysis, as emphasized by Nathanson:

1. Precipitation- and chaff-clutter spectra depend on a wind-shear effect, the result of change in wind velocity with altitude.
2. The average clutter velocity itself, as well as fluctuations about this average, imposes limitations on MTI effectiveness.
3. Clutter lying beyond the unambiguous radar range further degrades MTI performance.

B.2 TYPES OF CLUTTER REGIONS

For purposes of MTI processing, the environment is divided into clutter regions for both rain and chaff. The geometrical and physical properties of the regions are specified by the user under the constraints of the model. Three *geometrical configurations* of clutter regions are allowed:

*F. E. Nathanson, *Radar Design Principles*, McGraw Hill, New York, 1969.

Type 1 (band). This region is defined by specifying a lower and an upper height level above the Earth's surface. The type 1 clutter region lies everywhere over the Earth's surface between these two boundaries.

Type 2 (half-band). This region is defined by specifying a lower and an upper height level as in type 1, with the addition of a boundary line, determined by a point through which it passes, and a normal to the line pointing outward from the region. The type 2 clutter region lies over the portion of the Earth's surface between the two height boundaries that is to the side of the boundary line away from the normal.

Type 3 (cylinder). This region is defined by specifying a lower and an upper height level as in type 1, with the addition of the center and radius of a disk on the Earth's surface. The type 3 clutter region lies in the cylinder formed over the disk between the specified height levels.

B.3 TYPES OF CLUTTER-REJECTION FILTERS

MTIMOD accepts four types of clutter-rejection filters. The first three types are actually special cases of the fourth, more general, type, and are included to reduce input required for the simplest and most commonly used filters. The four filter types are as follows:

Type 1 (series of simple delay-line cancelers)

$$S(t) \rightarrow \sum_{k=0}^{N-1} \binom{N-1}{k} (-1)^k S(t + kT) .$$

A sequence of N consecutive signals $S(t)$, each separated in time by T, is summed by the filter using binomial coefficients.

Type 2 (generalized series of simple delay-line cancelers)

$$S(t) \rightarrow \sum_{k=1}^{N} A_k S[t + (k-1)T] .$$

A sequence of N consecutive signals $S(t)$, separated in time by T, is summed by the filter using weighting factors A_k, specified by the user.

Type 3 (delay-line cancelers with variable sampling rate)

$$S(t) \rightarrow \sum_{k=1}^{N} A_k S(t + B_k T) .$$

A sequence of N signals $S(t)$, the kth signal sampled at $B_k T$ after the first, is summed by the filter using weighting factors A_k. The sequences A_k and B_k are specified by the user.

Type 4 (delay-line cancelers with staggered PRF)

$$S(t) \to \sum_{k=1}^{N} A_k S\left(t + \sum_{j=1}^{M} B_{kj} T_j\right).$$

A sequence of N signals $S(t)$ is summed by the filter using weighting factors A_k. The M repetition intervals T_j combine to form the kth sample at $\Sigma_{j=1}^{m} B_{kj} T_j$. The sequences A_k and B_{kj} are specified by the user.

B.4 RAIN- AND CHAFF-CLUTTER INPUT

MTIMOD requires the current target position and velocity as calculated by SURSEM. The model has the capability to calculate a given number of target positions at equally spaced intervals along the target track and perform MTI signal processing at each position, if desired; however, in its present configuration, SURSEM will call MTIMOD individually for MTI processing at each target position it determines.

Given an actual target position, MTIMOD calculates a number of target "pseudopositions." These pseudopositions are locations on each radar antenna pattern sidelobe at the target range and at the second-time-around position corresponding to both the actual target position and the sidelobe positions. Calculations are performed for each target position (either real or pseudo, as defined above), and henceforth the general term "target position" will refer to any one of the 2(1 + number of sidelobes) positions described.

Each target position lies in at most one rain-clutter region and one chaff-clutter region. The rain- and chaff-clutter regions are taken to be distinct and are defined separately even if their geometric boundaries are the same.

For a given target position, MTIMOD determines the rain-clutter region, if any, and the chaff-clutter region, if any, in which the target lies. The rain-clutter power input to a clutter-rejection filter resulting from the target lying in the rain-clutter region is given by

$$\text{RPOWIN} = (3.7729 \times 10^{-27}) \left[\frac{PG^2 L(\Delta\theta)(\Delta\phi)\tau}{\lambda^2 R^2}\right] Z$$

where

P = peak pulse power (watts)

G = "equivalent gain" of the sidelobe (or main lobe) corresponding to the given position

L = lumped system losses

$\Delta\theta, \Delta\phi$ = the equivalent vertical and horizontal beamwidths (degrees)

τ = pulse width (microseconds)

λ = wavelength (meters)

R = range (kilometers)

Z = rainfall reflectivity factor, usually taken to be

$$Z = \text{RCOEF} \times \text{RRATE}^{\text{REXPO}}$$

where RRATE is the rainfall rate (millimeters per hour) in the selected rain-clutter region and RCOEF and REXPO are inputs, usually taken to be

$$\text{RCOEF} = 200.\,,$$

$$\text{REXPO} = 1.6\,.$$

The chaff-clutter power input to a clutter-rejection filter resulting from the target's lying in the chaff-clutter region is given by

$$\text{CPOWIN} = (1.3606 \times 10^{-8}) \left[\frac{PG^2 L(\Delta\theta)(\Delta\phi)\tau\lambda^4}{R^2}\right] DN\,,$$

where the parameters are the same as those defined for rain clutter and DN is the number of dipoles per unit volume (cubic meters) in the selected chaff-clutter region.

B.5 RAIN- AND CHAFF-CLUTTER OUTPUT FROM CLUTTER-REJECTION FILTER

A general clutter filter (type 4) has the form

$$S(t) \to \sum_{k=1}^{N} A_k S\left(t + \sum_{j=1}^{M} B_{kj} T_j\right)$$

and contains filter types 1 through 3 as special cases. For this general filter type the rain- or the chaff-clutter output (normalized for a clutter input of unit strength) is given by

$$\text{FACOUT} = \sum_{j=1}^{N} \sum_{k=1}^{N} A_j A_k \rho \left[\sum_{r=1}^{M} (B_{jr} - B_{kr}) T_r\right].$$

The function $\rho = \rho(x)$ is defined by

$$\rho(x) = \cos\left[4\pi \frac{V_0}{\lambda} x\right] \exp\left[-2\left(\frac{2\pi\sigma_v x}{\lambda}\right)^2\right]$$

where V_0 is the mean rain- or chaff-clutter velocity (meters per second), equal to zero for clutter locking, and σ_v^2 is the mean clutter velocity spread (meters per second):

$$\sigma_v^2 = \sigma^2(\text{shear}) + \sigma^2(\text{turbulence}) + \sigma^2(\text{beam})$$
$$+\sigma^2(\text{fall}) + \sigma^2(\text{antenna motion}) .$$

There are five independent mechanisms that contribute to σ_v^2, the variance of the rain or of the chaff clutter velocity, as indicated above.

Wind shear. The change in wind speed with altitude results in a distribution of radial velocities over the vertical extent of the beam, so that

$$\sigma(\text{shear}) = (5.20457 \times 10^{-3}) \times \begin{Bmatrix} \text{RAINGRAD} \\ \text{or} \\ \text{CHAFGRAD} \end{Bmatrix} \times \text{RANGE} \times \text{DELTHETA}$$

where (RAINGRAD or CHAFGRAD) is the wind-speed gradient measured at the lowest height level of the clutter region (5.7 m/sec-km nominal value), RANGE is the slant range to clutter (kilometers), and DELTHETA is $\Delta\theta$, the half-power vertical beamwidth (degrees).

Beam broadening. The finite width of the radar beam causes a spread of radial velocity components of the wind when the radar is looking crosswind, so that

$$\sigma(\text{beam}) = (5.20457 \times 10^{-3})\, V_0 \times \text{DELPHI} \times \sin\beta$$

where V_0 is the mean clutter velocity (meters per second), DELPHI is $\Delta\phi$, the half-power horizontal beamwidth, and β is the angle between the line-of-sight and wind-velocity vector.

Turbulence. Fluctuating currents of the wind cause a radial velocity distribution centered at the mean wind velocity:

$$\sigma(\text{turbulence}) = \begin{Bmatrix} \text{RAINTURB} \\ \text{or} \\ \text{CHAFTURB} \end{Bmatrix} \text{ (1.0 m/sec nominal value) .}$$

Fall velocity distribution. A spread in the fall velocities of the reflecting particles results in a spread of velocity components along the beam, so that

$$\sigma(\text{fall}) = \begin{Bmatrix} \text{RAINFALL} \\ \text{or} \\ \text{CHAFFALL} \end{Bmatrix} \times \sin(\text{elevation angle})$$

where (RAINFALL or CHAFFALL) is the mean fall velocity (1.0 m/sec nominal value).

Antenna rotation. The angular rotation of the antenna results in spectrum broadening of the clutter echoes, giving

$$\sigma(\text{antenna}) = (0.7897854) \times \text{AR} \times \lambda \times \frac{1}{\text{DELPHI}}$$

where AR is the antenna rotation rate (rpm), λ is the radar wavelength (meters), and DELPHI is $\Delta\phi$, the half-power horizontal beamwidth.

It is always assumed that the target and second-time-around pseudopositions are illuminated by the full number of pulses, N, required by the filter.

B.6 SIGNAL POWER OUTPUT FROM CLUTTER-REJECTION FILTER

For the general filter described earlier,

$$S(t) \to \sum_{k=1}^{N} A_k S\left[t + \sum_{j=1}^{M} B_{kj} T_j\right],$$

the signal power output (normalized for a signal input of unit strength) is given by

$$\text{SIGFAC} = \left|\sum_{j=1}^{N} A_j \exp\left(2\pi i f_d \sum_{k=1}^{M} B_{jk} T_k\right)\right|$$

where f_d is the doppler frequency of the target.

B.7 OUTPUT

The output from MTIMOD consists of the total (rain and chaff) clutter power input to each clutter-rejection filter, the total (rain and chaff) clutter power output from each clutter-rejection filter, and the signal/power output from each clutter-rejection filter, for each specified actual target position. At present the model will accept up to 50 rain- and 50 chaff-clutter regions each of types 1, 2, and 3. A total of 10 clutter-rejection filters may be defined for a run. Once the clutter regions and filters have been defined and an actual target position specified, MTIMOD determines the corresponding target pseudopositions described earlier; together with the actual position, these pseudopositions become the set of positions for which the individual MTI calculations are made.

For each clutter-rejection filter to be considered MTIMOD computes the following output, as outlined above, by actual target position. Let

NSL = number of antenna pattern sidelobes,

J = sidelobe "pseudoposition" ($J = 1$ for main-lobe true position),

I = 1 for the first-time-around position,

I = 2 for the second-time-around position.

Thus $\{(I, J) | 1 \leq I \leq 2,\ 1 \leq J \leq (1 + \mathrm{NSL})\}$ indexes the set of pseudopositions. Recall that

RPOWIN(I,J) = rain-clutter power input to filter from (I, J)th pseudoposition,

CPOWIN(I,J) = chaff-clutter power input to filter from (I, J)th pseudoposition.

The total rain-clutter and total chaff-clutter inputs from all pseudopositions are

$$\mathrm{RTOTIN} = \sum_J \sum_I \mathrm{RPOWIN(I,J)}\ ,$$

$$\mathrm{CTOTIN} = \sum_J \sum_I \mathrm{CPOWIN(I,J)}\ .$$

Hence the *total clutter* input to the filter due to a target at the specified position is

$$\mathrm{TOTIN} = \mathrm{RTOTIN} + \mathrm{CTOTIN}\ .$$

Let

RFAC(I,J) = filter output from rain-clutter input due to (I, J)th pseudoposition, normalized for unit power input,

= $\mathrm{FACOUT}_{(\mathrm{rain})}$ as already defined,

and

CFAC = filter output from chaff-clutter input due to (I, J)th pseudoposition, normalized for unit power input,

= $\mathrm{FACOUT}_{(\mathrm{chaff})}$ as already defined.

The total rain-clutter and total chaff-clutter filter outputs from all pseudopositions are

$$\mathrm{RPOWOUT} = \sum_J \sum_I \mathrm{RPOWIN(I,J)} \times \mathrm{RFAC(I,J)}\ ,$$

$$\mathrm{CPOWOUT} = \sum_J \sum_I \mathrm{CPOWIN(I,J)} \times \mathrm{CFAC(I,J)}\ ,$$

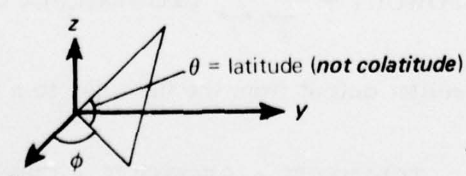
and hence the *total clutter* output from the filter due to a target at the specified position is

$$\mathrm{TCLUTOUT} = \mathrm{RPOWOUT} + \mathrm{CPOWOUT}\ .$$

The signal power output, normalized for a signal input of unit power, from the clutter-rejection filter is given as SIGFAC, which has already been defined.

The required inputs from SURSEM to MTIMOD are listed in Table B.1-1, and the MTIMOD macro flowchart is shown in Fig. B.1-2.

Table B.1-1 — Inputs from SURSEM to MTIMOD

Symbols (in MTIMOD)	Description	Dimension
NUMBER	Number of target positions	Integer
INCRE	Time increment between target positions	s
TCS	Target cross section	m^2
XT, YT, ZT	Target coordinates	km
XR, YR, ZR	Radar coordinates	km
VXT, VYT, VZT	Target velocity components	m/s
VXR, VYR, VZR	Radar velocity components	m/s
P	Peak pulse power	W
FRE	Carrier frequency	Hz
TAU	Pulse width	μs
LOSS	Lumped system losses occurring in radar range equation	–
WX, WY	Wind-speed components at height *H*	m/s
H	Height	km
MBP	Elevation of main beam	deg
BWVERT	One-way 3-dB vertical beamwidth (main beam)	deg
BVHOR	One-way 3-dB horizontal beamwidth (main beam)	deg
GAIN	Gain	dB
NSL	Number of sidelobes	Integer
(FOR 1 ⩽ J ⩽ NSL) THETA(J), PHI(J)	Position of *J*th sidelobe with respect to main-beam axis	deg
G(J)	Gain of *J*th sidelobe	dB
DELTHETA(J), DELPHI(J)	Vertical and horizontal beamwidths of *J*th sidelobe	deg
		

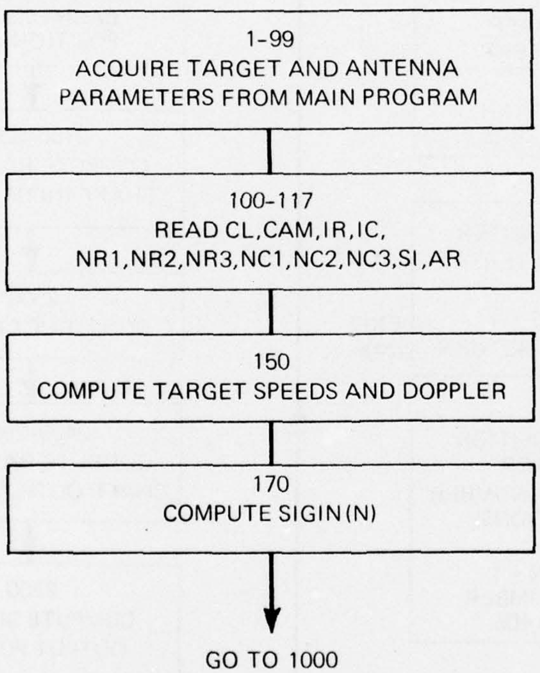


Fig. B.1-2—MTIMOD macro flowchart

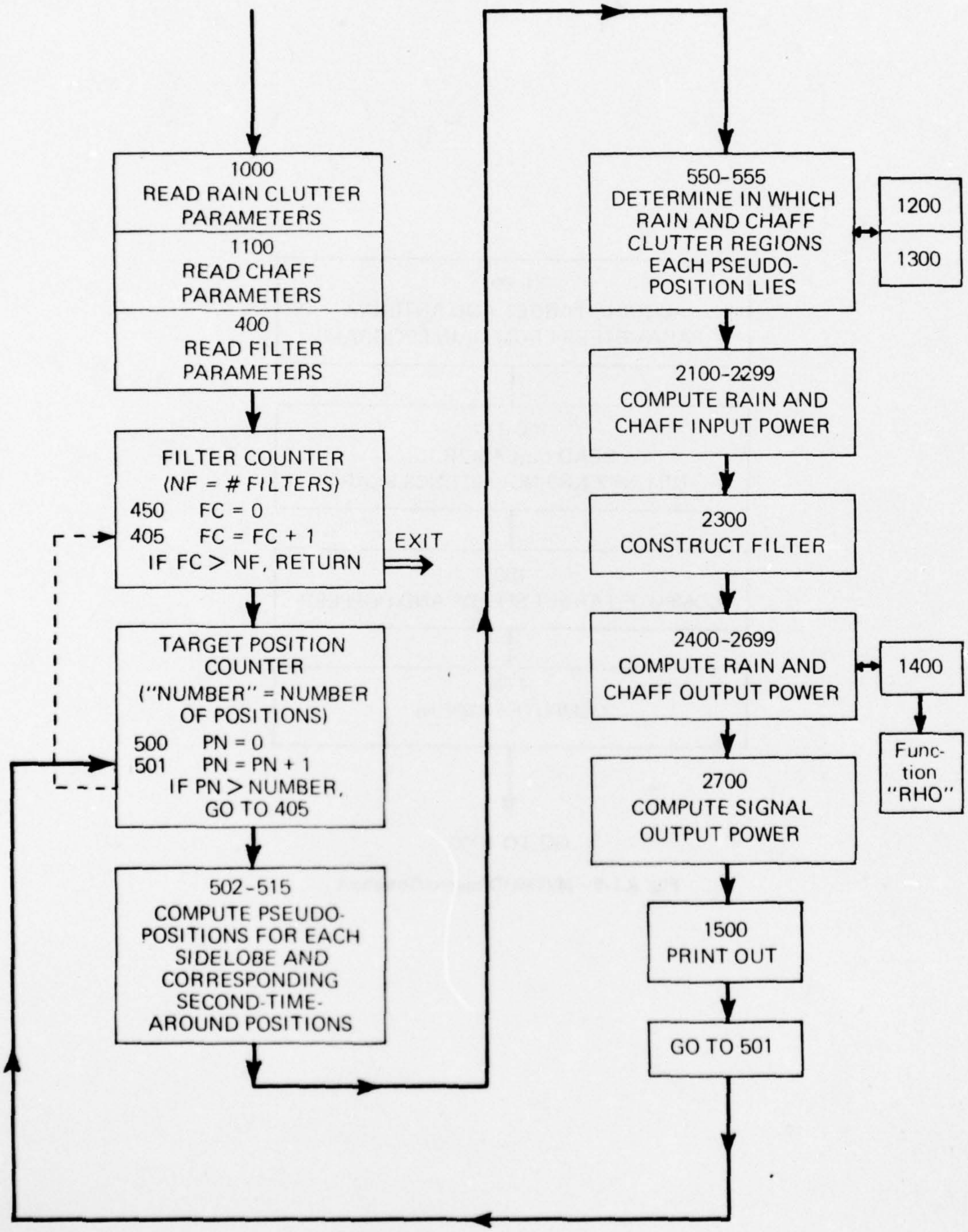


Fig. B.1-2—MTIMOD macro flowchart (Continued)